

IP-CONNECSY V5.5

for
Avaya Definity® & Communication Manager

Technical Manual



IP-CONNECSY V5.5

PC Console for Avaya Definity® & Communication Manager

Technical Manual

**Versie 5.5/Plus UK
September 2010**

Avaya Nederland BV

Produced and Distributed by **Pridis**

Contents

CONTENTS.....1

1. REQUIREMENTS.....5

1.1 HARDWARE 5

1.2 SOFTWARE.....5

2. FEATURES 7

2.1 USER-INTERFACE 7

2.2 DATABASE.....7

2.3 FUNCTIONALITY 7

3. CONNECSY INSTALLATION.....8

3.1 SOFTWARE INSTALLATION 8

3.2 DATABASE INSTALLATION (LOCAL OR NETWORK) 8

3.3 DEFAULT INSTALLATION 9

3.4 CONNECSY INSTALLATION STEP BY STEP 10

3.4.1 Starting the installation 10

3.4.2 The welcome screen and License Agreement 11

3.4.3 Components 12

3.4.4 Set SQL Server login credentials.....12

3.4.5 Choose destination location 13

3.4.6 Borland Database Engine 13

3.4.7 Completing the installation 14

3.4.7 License Activation 15

3.4.8 Installing Connecsy for SQL Server & Installing the Data Service 1520

3.4.9 Configuring the Data Service 222

4. CLIP INFORMATION 244

4.1 GENERAL 244

4.2 CONFIGURE CLIP INFORMATION 25

4.2.1 Search for Clip information.....25

4.2.2 Field to use for clip info 25

4.2.3 Area prefix, Country prefix and Dail prefix 26

5. DXS/BLF INFORMATION 26

5.5 GENERAL 26



5.2 CONFIGURE DXS/BLF INFORMATION	27
5.2.1 DXS/BLF range settings	288
5.2.2 DXS form captions.....	28
5.2.3 DXS/BLF range (row and column) settings.....	29
5.2.4 DXS/BLF range (number mapping table) settings	29
5.2.5 DXS/BLF range (name mapping table) settings	29
6. CONNECSY DATABASE.....	30
6.1 GENERAL	30
6.2 DATABASE INTERFACE	30
6.3 BDE CONFIGURATION	31
6.4 ODBC DRIVER.....	32
6.5 CONNECSY DATABASE INI SETTINGS	33
7. THE FIREBIRD DATABASE ENGINE	35
7.1 THE CLIENT SOFTWARE.....	35
7.2 THE SERVER ENGINE	35
7.3 CONTROLLING THE SERVER ENGINE.....	35
7.4 CONFIGURING THE SERVER.....	36
7.5 CONNECTING TO A DATABASE	36
7.5.5 Connecting to a database from BDE	36
8. THE SQL SERVER DATABASE ENGINE.....	38
8.1 THE CLIENT SOFTWARE.....	38
8.2 THE SERVER ENGINE	38
9. CONNECSY IMPORT/EXPORT	49
9.1 CONNECSY IMPORT	49
9.2 COMMA SEPARATED VALUE (CSV-FILE)	49
9.3 CONNECSY EXPORT	49
9.4 CONNECSY AUTOMATIC IMPORT	50
9.4.1 Day of import	50
9.4.2 Starttime of automatic import.....	51
9.4.3 Import and error log file.	51
9.4.4 Connecsy Automatic Import (Advanced).....	51
9.5 COMMA SEPARATE VALUE (CSV) WITH ID.....	52
9.6 IMPORT / DOWNLOAD DEFINITY® FILE.....	53
10. CONNECSY SOUNDS	55
10.1 SOUND TYPES	55
10.2 REMINDER ON HOLD TIMER FOR “HELD CALL RINGING”	56
10.3 WAITING CALL RINGING	56
10.4 SOUND QUEUE RINGING.....	56



11. CONNECSY QUEUE INFORMATION.....	58
11.1 CONFIGURING QUEUE INFORMATION	58
11.1.1 <i>Definity Queue Types</i>	58
11.1.2 <i>Definity Queue names</i>	59
11.1.3 <i>Definity Queue String Identifier</i>	59
11.1.4 <i>Connecsy Poll-rate</i>	59
11.1.5 <i>Put line on hold before transfer dial</i>	60
11.1.6 <i>DEFINITY® uses attendant vectoring</i>	60
12. PRESENCE DATA SERVICE	61
12.1 GENERAL	61
12.2 SINGLE ATTENDANT SETUP	61
12.3 AUTOMATICALLY ADD EXTENSIONS TO CONNECSY	61
13. CONNECSY CONFIGURATOR	63
13.1 CONNECSY LANGUAGE SELECTION	63
13.2 VERSION CODES.....	64
13.3 COMMON SETTINGS.....	64
13.3.0 <i>Title</i>	65
13.3.2 <i>Automatic Clear</i>	65
13.3.3 <i>Cursor position</i>	65
13.3.4 <i>Automatic sort</i>	66
13.3.5 <i>Display Alphanumeric keyboard</i>	66
13.3.6 <i>DXS Simulation</i>	66
13.4 COMMUNICATION INTERFACE.....	67
13.4.1 <i>Communications link to DEFINITY®</i>	67
13.4.2 <i>DEFINITY® database import (Port and Baud rate)</i>	69
13.5 DATABASE.....	69
13.6 LOGON.....	70
13.7 FEATURE BUTTONS	70
13.8 FUNCTION BUTTONS	72
13.8.1 <i>Caption and Hints of the function keys</i>	72
13.8.2 <i>Functions key bitmaps (file)</i>	72
13.8.3 <i>Programmable function buttons</i>	73
13.9 SEARCH BUTTONS	73
13.9.1 <i>Search button captions and hint</i>	74
13.9.2 <i>Location of database fields</i>	74
13.9.3 <i>Edit field search definition</i>	75
13.9.4 <i>Calendar</i>	75
13.9.5 <i>Email</i>	75
13.10 SHEET CONTENTS.....	75
14. HAND MADE CONFIGURATIONS.....	77

14.1 SCROLL BUTTON BITMAPS	77
14.2 F12 SEARCH BUTTON.....	77
14.3 ARROW KEY EDIT FIELD ORDER	77
14.4 CHARACTER MAPPING TABLE	78
15. DEFINITY® SETTINGS	79
15.1 ATTENDANT TYPE AND AUTO ANSWER	79
15.2 DXS/BLF	79
15.3 FEATURE BUTTONS	80
15.4 AUTO START.....	81
15.5 CALLING LINE IDENTIFICATION	81
15.6 TIME REMAINDER ON HOLD	82
15.7 HOLD-AUTOMATIC.....	82
15.8 EXTENSION PORT TYPE	82
15.9 LICENSES.....	83
15.10 ENABLING SNMP	83
16. CONNECSY PLUS	86
16.1 AUTOMATIC RECORDING	86
16.2 MANUAL RECORDING	87
16.3 PLAYING RECORDED FILES.....	88
16.4 INSTALLATION	88
17. CONNECSY LDAP	91
17.1 ACTIVATING LDAP FUNCTIONALITY	91
17.2 MANUAL IMPORT.....	91
17.3 AUTOMATIC IMPORT	91
17.4 LDAP PROFILE-FILES	91
<i>[General]</i>	91
<i>[LDAP Session]</i>	91
<i>[Mapping]</i>	92
<i>Queries</i>	92
<i>Database fields</i>	92
<i>\$SPLIT</i>	93
<i>\$CONCAT</i>	93
<i>\$LIST</i>	93
<i>\$VALUE</i>	94
<i>Nested directives</i>	94

1. Requirements

1.1 Hardware

Connecsy PC:

- Minimal: Pentium4®. Recommended: Intel Pentium4 1.5GHz or AMD AthlonXP 1500+ or faster
- Minimal: 1024Mb for Windows 7, Vista or XP.
- Minimal 1 GB of free disk space required
- Minimal: SVGA compatible video card with a resolution of 800x600, 256 colours. Recommended: SVGA compatible video card with a resolution of 1024x768 or higher and true colour display
- Serial adapter (required when using serial DCP for communication with Callmaster VI or PassageWay®)
- Serial adapter (optional - communication touch screen)
- Network card (required when using the H.323 IP-Attendant protocol), recommended: 100Mbit Ethernet card
- DirectSound 8 or 9 compatible sound card (WDM drivers recommended)

For Connecsy Plus:

- Sound card (full-duplex)
- Digital Speech Converter

Connectivity:

- Direct VOIP (H.323) connection with or without separate IP Phone
- Callmaster VI (only when using serial DCP)
- HW 41 (one of the following devices: 8411D or CallMaster VI)

PABX:

- DEFINITY® or Communication Manager
- C-LAN card required when using the H.323 IP-Attendant protocol

1.2 Software

Operating System PC

- Windows™ 7
- Windows™ XP Professional with Service Pack 3 or higher

DirectX version 9 or higher

Connecsy

- IP-Connecsy version 5.5 or newer
- Build 14 or newer

PassageWay® (only when using serial DCP)

- SW 01



DEFINITY®

- Version 4 (minimal. Version 7 required for BLF display of Send Calls. Version 10 or higher required when using the H.323 IP-Attendant protocol)

2. Features

2.1 User-interface

keyboard, mouse and touch-screen compatible
configurable screen layout
configurable button text
language independent
use of different colours to easily distinguish the line status
optional use of bitmaps indicating line status

2.2 Database

database can either be locally or LAN-based
open database connectivity (ODBC), connect to your own database
import/export functionality
import from a LDAP enabled data directory, for example Windows™ Directory Services
optional dynamic link to your Communication Manager® PABX
12 database search fields
configurable search algorithm (search more than one field)
optional search add-ons (wild-cards, character mapping)
separate database entry module available (no Phone or CTI and no Dongle required)

2.3 Functionality

Attendant Console (incl. feature buttons)
DXS/BLF console
DXS/BLF console hundred group switching depending on search results
DXS/BLF Logged of stations (IP and digital stations)
optional 'qcalls' (hunt groups queue) information (graphical display)
optional 'atd-qcalls' (attendant queue) information (graphical display)
optional clip information displayed in the line buttons

3. Connecsy Installation

3.1 Software installation

On receipt of your Connecsy package, you must check if it contains the following items:

- Connecsy installation CD-ROM,
- Connecsy protection dongle,
- Connecsy license information card.

Contact your vendor if one or more of these items are missing.

Connecsy must be installed using the installation CD-ROM provided by the vendor. The software can either be installed on a local machine or on a network. The Connecsy user will not notice the difference in using Connecsy, however installation-wise there are some minor differences.

Regardless from the type of install you perform, some components needed by Connecsy need to be installed locally. These components are:

Borland Database Engine (BDE),
Microsoft SQL Server Express Edition or Firebird SQL database engine software,
System DLL's and controls,
DirectX,
Dongle driver.

This will require you to run the setup procedure, regardless if you're doing a local or network installation. DirectX is not installed by the setup procedure, a separate install for the latest DirectX version is provided on the installation CD-ROM however.

The software can either be installed locally on the PC or on a network drive by changing the destination location during the setup procedure. By default (recommended) the software is installed on the local machine. Installing the software on the local machine does not necessarily mean that the database has to be on the local machine. Although the software is installed locally, the databases can be located on a remote machine.

By default the Connecsy databases are installed into the subdirectory \DBS of the Connecsy application directory.

Important: The Connecsy application is linked to the databases by an alias. The BDE provides these aliases.

More information about BDE will be available later in this manual.

3.2 Database installation (local or network)

Connecsy 5.5 supports any ODBC compatible database engine, but to make life a little more simple two database engines are described in this manual: Microsoft SQL Server Express and Firebird 2.1. The latter is provided on the CD. Before installation one should consider which one will be selected to be used. This mostly depends on ICT requirements.

The MS SQL Server Express will require some more manual steps to finish installation. Also for installations with more than 3000 records, the MS SQL Server Express is preferred due to a better performance.

By default Connecsy 5.5 is installed with the Firebird 2.1.2.18118 database engine. Firebird is a client/server database engine. All machines that need to run Connecsy are required to have the Firebird client software installed locally. The same does apply to the BDE engine. If you decide to host the Connecsy database locally, the Firebird server engine software needs to be installed locally. If you decide to host the Connecsy database on a remote machine, the Firebird server engine needs to be installed on the remote machine. Installation of the Firebird server engine requires a user logon with administrative privileges. If you choose to host the databases on a remote machine, you'll need to change the path reference of the database alias inside the BDE engine (see BDE configuration in chapter 'Database'). This is also required if you install Connecsy to another path then the default path or your Program Files folder is not located at C:\Program Files.

You will get better performance if you select to host the databases on the local machine. However, if you have multiple Connecsy's changes to the data need to be synchronized over all the databases. Synchronization is possible using the 'automatic import procedure' available in Connecsy. For instance a file containing the last data can be imported each night and automatically previous imported records in the database will be removed. Records locally added to the database will never be removed during an import, thus creating a situation where there is a difference in imported records (central database) and locally added records (personal database). Additionally, 3rd party tools are available to automatically synchronize two or more Firebird databases (e.g. Borland IB-Replicator). More of the import and export is described in chapter 'Connecsy Import/Export'.

The installation of Microsoft SQL Server Express is described in detail in chapter 8 of this manual.

3.3 Default installation

The Connecsy V5.5 setup procedure contains the necessary modules as well as database components required to run Connecsy. After installation, Connecsy can be started using the default configuration settings; customisations must be done using the setup tool. The following files will be copied to the destination directory:

Connecsy software modules

\\Program Files\Pridis\Connecsy5.5\ directory

Connecsy databases components

\\Program Files\Pridis\Connecsy5.5\Dbs' directory

Connecsy language configuration files

\\Program Files\Pridis\Connecsy5.5\Deu' directory

\\Program Files\Pridis\Connecsy5.5\Enu' directory (engels)

\\Program Files\Pridis\Connecsy5.5\Esp' directory

\\Program Files\Pridis\Connecsy5.5\Fra' directory

\\Program Files\Pridis\Connecsy5.5\Ita' directory

\\Program Files\Pridis\Connecsy5.5\Nld' directory

\\Program Files\Pridis\Connecsy5.5\Xxx' directory

`\Program Files\Pridis\Connecsy5.5\Yyy' directory

Connecsy bitmaps

`\Program Files\Pridis\Connecsy5.5\Bmps' directory

Connecsy sounds

`\Program Files\Pridis\Connecsy5.5\Wav' directory

Connecsy working directory

`\Program Files\Pridis\Connecsy5.5\Temp' directory

Borland Database Engine components

`\Program Files\Common Files\Borland Shared\BDE' directory

Connecsy Data Service components

`\Program Files\Pridis\Connecsy Dataservice\' directory

Firebird database engine components

`\Program Files\Firebird\Firebird' directory

MS SQL Express database engine components

`\Program Files\SQL Server' directory

The Connecsy V5.5 setup procedure also creates a number of icons for easy access to the different modules:

Desktop Icons

Connecsy V5.5

Program Folder Icons

Connecsy V5.5

Connecsy V5.5 demo

Connecsy V5.5 data

Connecsy V5.5 configurator

3.4 Connecsy installation step by step

3.4.1 Starting the installation

Insert the installation CD-ROM into the CD-ROM drive. If autoplay has been enabled for your system, the Connecsy Product Menu should appear. If it doesn't open, open My Computer and double-click the icon for your CD-ROM drive. You can also right-click on the icon and select Autoplay from the context menu. You should be presented with the Connecsy Product Menu now (see figure 3-1).

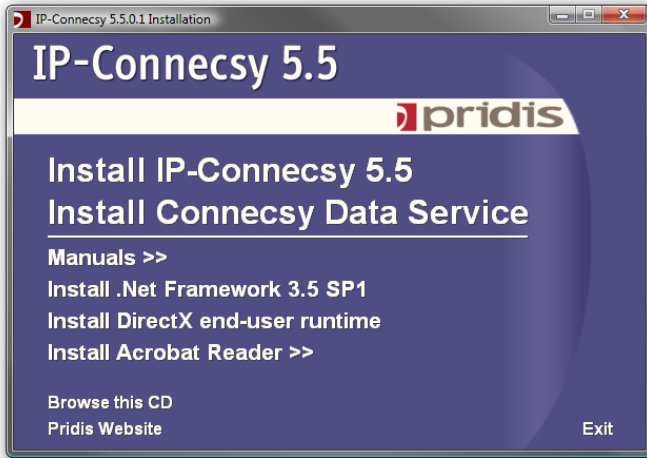


Figure 3-1: Connecsy Product Menu

The Connecsy Product Menu enables you to start the Connecsy installation, read the online manuals on the CD-ROM (requires Adobe Acrobat Reader, which can be installed from the CD-ROM using the product menu), install DirectX, browse the CD-ROM or browse the Pridis website on the internet. To start installing Connecsy, click Install Connecsy 5.5 in the product menu. This will start the InstallShield Wizard for Connecsy 5.5

Note: The InstallShield Wizard requires a logon with administrative privileges on the system. If the InstallShield Wizard complains about your privileges, logon with an account with administrative privileges on the system or ask your system administrator to do so.

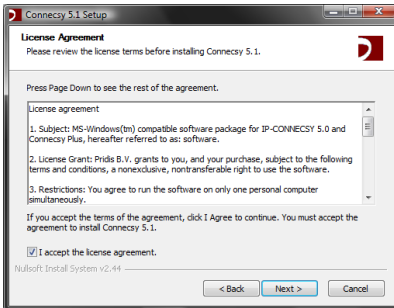
During initialization, the installer will check if there's a Firebird server running on the machine. If it finds one, the installer will display an error dialog. Be sure to stop the server before continuing (see the chapter about Firebird on how to do this).

3.4.2 The welcome screen and License Agreement

After the InstallShield Wizard has been initialized, the Welcome screen will appear. Click **Next** to continue.



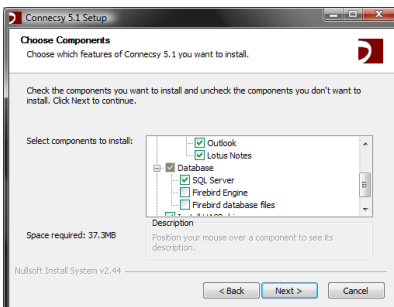
Click **Yes** to accept the License Agreement. If you refuse to accept the License Agreement, the installer will abort without installing Connecsy.



3.4.3 Components

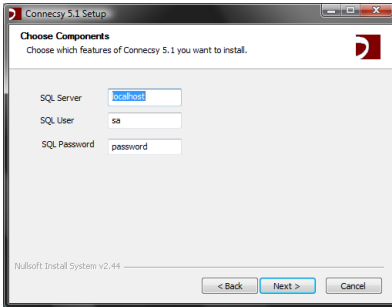
You can deselect one of the below components:

- Mediaconnectors
Serial or Avaya IP
- Calendar integration
Outlook or Lotus Notes
- Database engines
SQL Server or Firebird
- Hasp Dongle drivers



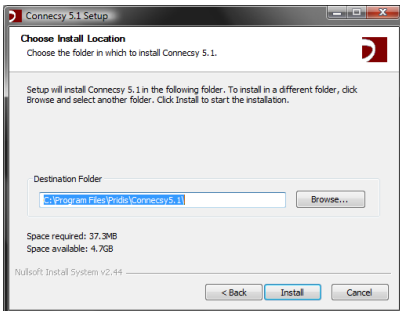
3.4.4 Set SQL Server login credentials

When the SQL Server has been selected, the system name and password to access the SQL Server has to be set. Leave the SQL Server host entry to *localhost* if SQL Server is installed on the same system as Connecsy. Otherwise enter the IP address of the server where SQL Server resides.



3.4.5 Choose destination location

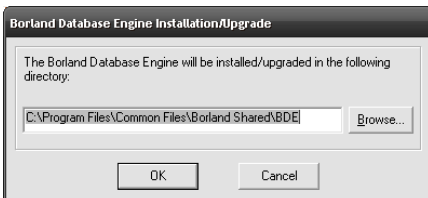
By default, the installer will install the software inside the C:\Program Files\Pridis\Connecsy folder. If you want to install the software inside another folder, click **Browse...** and select the desired folder. Click **Next** to continue.

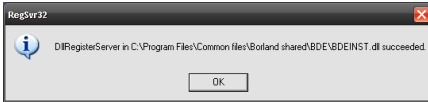


The installation will start installing all components.

3.4.6 Borland Database Engine

When the Firebird database has been selected, the following additional steps will be followed to install the BDE.





3.4.7 Completing the installation

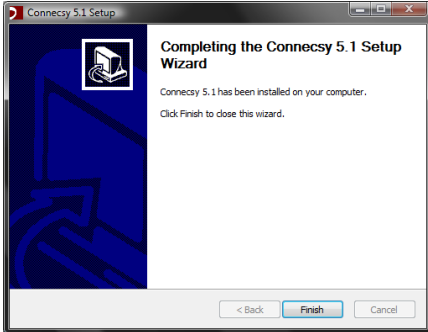


Figure 3-2: Finishing the installation.

3.4.7 License Activation

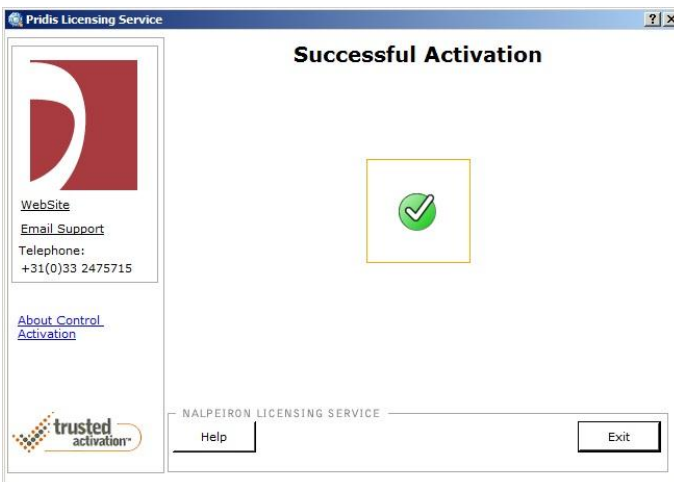
After installing Connecsy 5.5, you have to activate the license before the application will start. Go to the folder: *C:\Program Files\Pridis\Connecsy5.5* and open the *ControlActivation.exe*.



Click Continue.



Insert the License code and click 'Activate'.



The Connecsy 5.5 software is now activated. Click 'Exit' and start the Connecsy software.

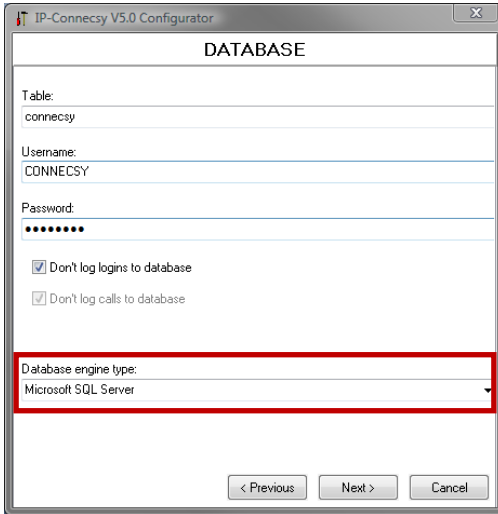
3.4.8 Installing Connecsy for SQL Server

When you have selected SQL Server as the main database engine, the following steps should additionally be followed to setup the connection between Connecsy and the SQL Server databases. This connection consists of installing the ODBC driver and configuring ODBC and the Borland Database Engine (BDE). See chapter 6 for a functional description on how this works together.

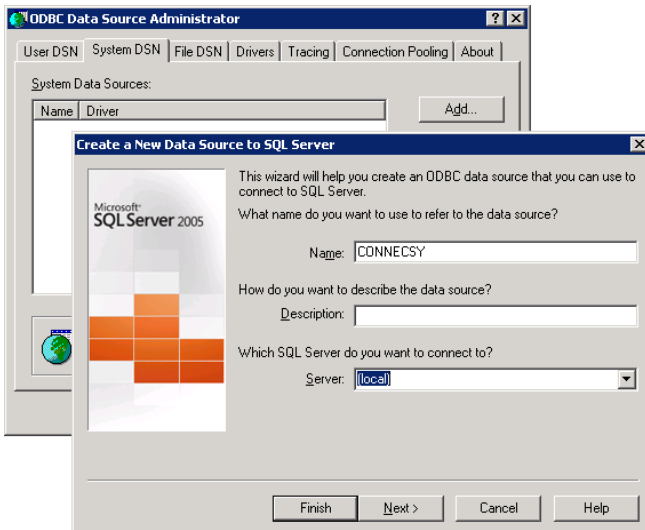
The ODBC driver is normally a Vendor-supplied driver. The ODBC driver can be installed and configured using the ODBC Administrator, which is located in the 'Control Panel'. In

some cases the driver must be added to the system, or in the case that the driver is already installed, it will only need some configuration changes.

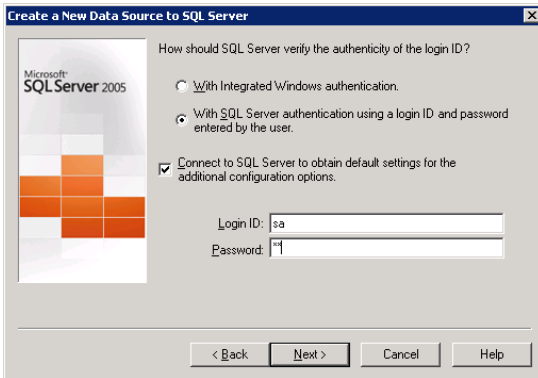
1. Open Connecsy Configurator and set the 'databasetype' to SQL Server within the DATABASE page.



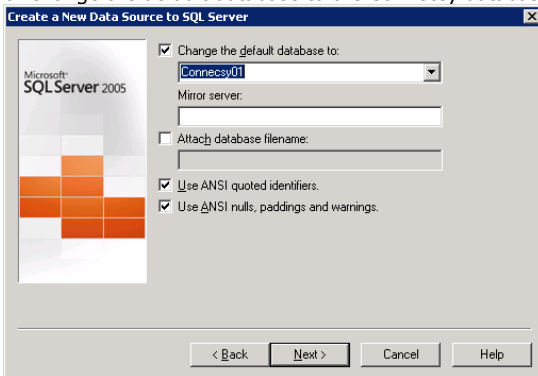
2. Open the ODBC Administrator and select the system DSN tab. Click add
3. Select the SQL Native Client, click Finish
4. In the Wizard, enter CONNCSY as DSN name and select local as server. Click Next



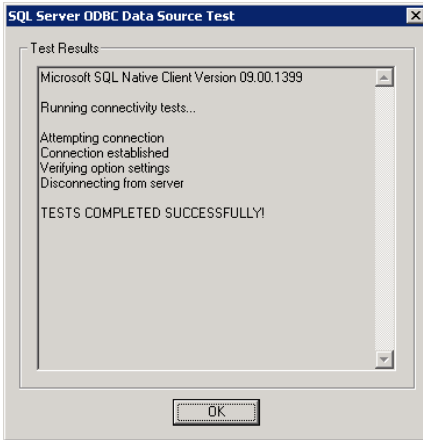
5. Change the login for this configuration



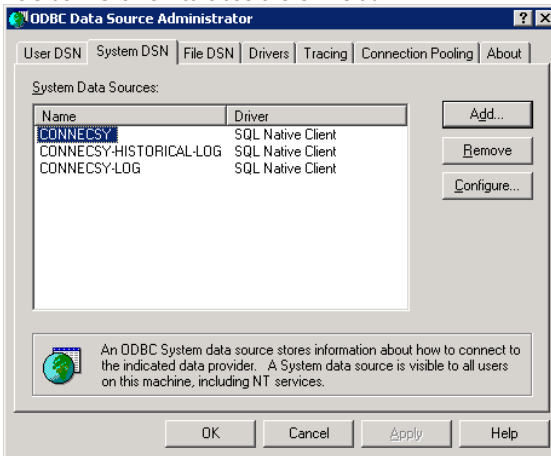
6. Change the default database to the Connecsy database, click Next



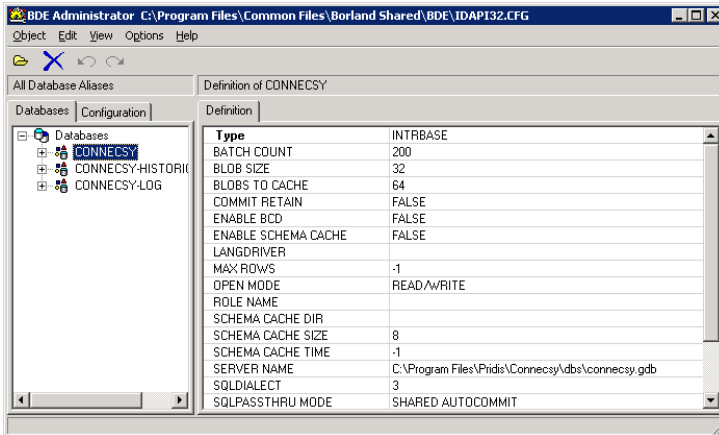
7. Click Next, then Finish. A popup summary box appears. Click Test to check the connection.



8. Repeat these steps to create the CONNECSY-LOG dsn and the CONNECSY-HISTORICAL-LOG dsn. Click OK to close the ODBC admin.



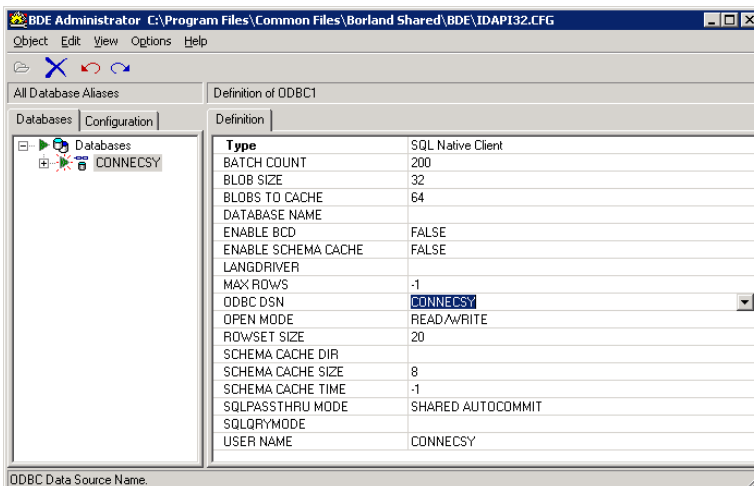
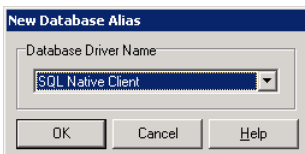
9. Open the Configuration Panel | BDE Administrator (or via *Program files|Common Files|Borland Shared|BDE|BDEAdmin.exe*)



10. Remove or rename any existing Connecsy database aliases.

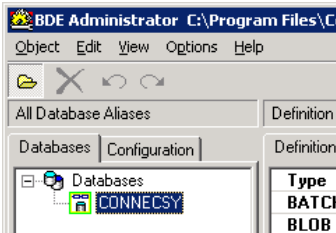
11. Right click the Databases-tab and select New...

12. Select the SQL Native Client, enter CONNECSY as alias name, type CONNECSY in the DSN field, and type CONNECSY as user name.



13. Right click the new alias and select Apply, and confirm the same with OK.

14. To test the connection doubleclick the alias and enter connecsy as password. A green rectangle indicates a correct connection. Rightclick the alias and select Close.



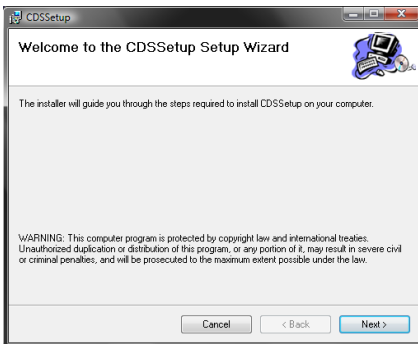
15. Perform steps 11 up to 14 for the two other database connection CONNECSY-LOG and CONNECY-HISTORICAL-LOG.

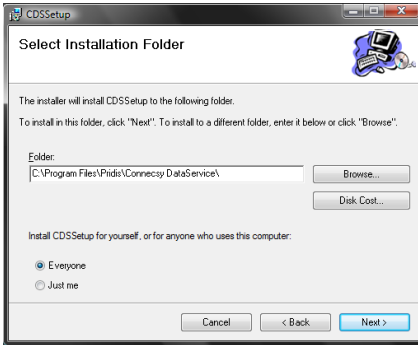
3.4.8 Installing the Data Service

When using the BLF information to get an indication of presence of a person, it is very useful to have information whether persons are logged on or off their telephone sets. This feature is provided within Connecsy by the Connecsy Data Service which in turn obtains its information from Avaya Communication Manager via a SNMP connection. This section describes the installation and configuration of the Data Service.

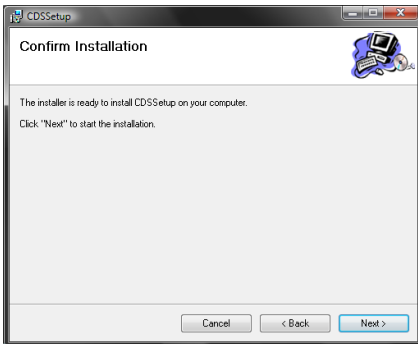
Note: *Be sure to complete the configuration steps within the Avaya Communication Manager first as described in detail in chapter 14.10, before continuing the next steps.*

After the Install Wizard has been initialized, the Welcome screen will appear. Click **Next** to continue.

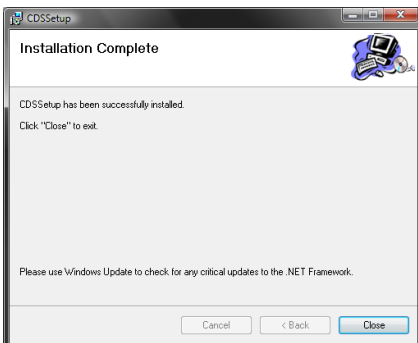




By default, the installer will install the software inside the C:\Program Files\Pridis\Connecsy DataService folder. If you want to install the software inside another folder, click **Browse...** and select the desired folder. Click **Next** to continue.



Confirm the installation and click **Next** to start installing all components.



3.4.9 Configuring the Data Service

The Connecsy Data Service fetches status information from one or more Communication Manager systems via Snmp and stores this information into the Connecsy database. Each of these connections should be configured.

Open Notepad or equivalent texteditor and open the file *C:\Program Files\Pridis\Connecsy DataService\ConnecsyDataService.exe.config*

Connections to Presence providing systems

Avaya Communication Manager

To configure the connection to Avaya Communication Manager find the section starting with <appSettings> and edit the setting 'SnmpHost1' as shown below and set the IP address of Communication Manager replacing the default address.

```
<appSettings>

    <add key="SnmpHost1" value="192.168.151.198" />

</appSettings>
```

For an additional Communication Manager system to be monitored, add a new line for 'SnmpHost2' and set the value to the IP address of this second system. Repeat this step for all Communication Manager systems.

In the case of a redundant or failover Mediasystem setup with double mediaservers (2x S87xx), there will be three IP addresses. The management IP address and the two physical IP addresses. This situation is configured using three SnmpHost references:

```
<appSettings>

    <add key="SnmpHost1" value="192.168.151.1" />
    <add key="SnmpHost2" value="192.168.151.198" />
    <add key="SnmpHost3" value="192.168.151.199" />

</appSettings>
```

Within the media server, the setting snmpCommunity is private by default. If anything else is set on the media server then this value should be adjusted in the config file:

```
<appSettings>

    <add key="snmpCommunity" value="private" />

</appSettings>
```

Connection to the Connecsy Database

For the connection to the Connecsy database, the settings depend if Firebird or SQL Server is used as database engine.

SQL Server Configuration:

```
<appSettings>

    <!--DBVersion 0=SQL Server (default), 1=Firebird-->
    <add key="DBVersion" value="0"/>
    <add key="DBCConnectionString" value="Data Source=localhost;Initial
Catalog=Connecsy;Persist Security Info=True;User ID=connecsy;Password=connecsy;" />
</appSettings>
```

Firebird Configuration:

```
<appSettings>

    <!--DBVersion 0=SQL Server (default), 1=Firebird-->
    <add key="DBVersion" value="1"/>
    <add key="DBCConnectionString"
value="Server=localhost;User=connecsy;Password=connecsy;Database=C:\Program
Files\Pridis\Connecsy5.0\dfs\CONNECSY.GDB" />
</appSettings>
```

Other Preferences

The speed of fetching and updating all the presence information in Connecsy may affect the performance and responsiveness of Connecsy for the enduser. The default setting, update each 3 seconds, is optimized for performance but still offering near real-time presence information. This interval can be altered by changing the default value in the configuration file. The value is set in milliseconds (3 sec = 3000 milliseconds):

```
<appSettings>

    <add key="TimerInterval" value="3000" />

</appSettings>
```

Save the changes to the file and restart the Data Service by using the Windows Services panel.

4. Clip Information

4.1 General

Clip information is information that can be displayed in the line buttons instead of using the default line information such as 'active' and 'on hold' etc. Normally the display string contains name and or number information of the person calling to the attendant console - this information is extracted from the display string and placed in the line button (see figure 4-1).

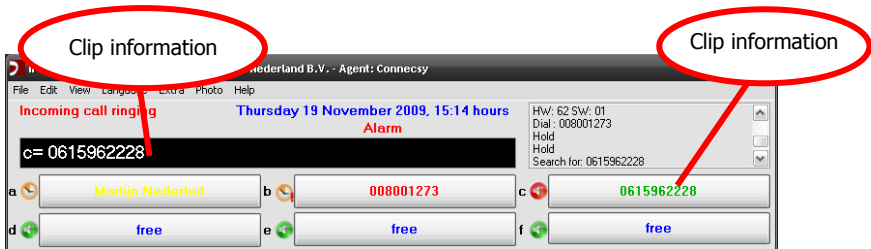


Figure 4-1: line button for Clip information

In some cases the display string contains a number, this number can be used to search in the database and when a record is found the name instead of the number will be displayed in the line button. The Connecsy operator will therefore see in the line button who is calling (by name) before the person has identified himself.

The implementation in Connecsy allows configuring maximum of three search fields in the database. The reason for this is to allow a search on not only extension but also for instance on cellular and car phone numbers. Since the database probably will not contain the string that is extracted from the display, add-ons can be used to make sure Connecsy will find the appropriate record anyway.

Why would we be using add-ons on the information extracted from the display is something that can be best explained using an example. For example the following call appears in the Connecsy display (let assume this is a supplier of something that is contacted regularly):

a = 3234441777 to Avaya Id.

The '3234441777' number is extracted from the display, obviously this number would never be found in the database since the Connecsy database would also contain the dial out prefix and or country and area prefix. Let's assume this supplier is located in Belgium and therefore be entered in the database as 0003234441777 (first zero dial out prefix, second/third zero country dial out prefix).

The 'Connecsy.INI' file allows you to specify the prefix strings so a search can be done on the following combinations:

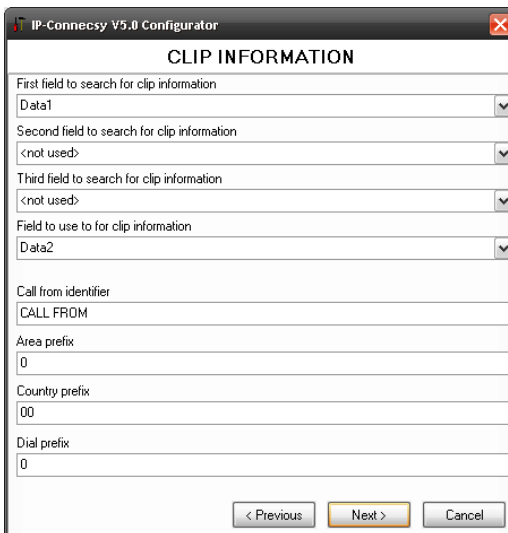
```

▪   ` 3234441777 '
<dial out prefix> '3234441777'
<dial out prefix><area prefix> '3234441777'
<dial out prefix><country dial out prefix> '3234441777'
<dial out prefix><country dial out prefix><area prefix> '3234441777'

```

4.2 Configure Clip Information

You can configure the clip information with the Connecsy Configurator at the 'CLIP INFORMATION' form shown in figure 4-2. We just described how clip information in Connecsy works. All the settings displayed in the picture below are set in the 'connecsy.ini' file in the main Connecsy directory.



The screenshot shows a window titled "IP-Connecsy V5.0 Configurator" with a close button in the top right corner. The main area is titled "CLIP INFORMATION". It contains several fields and buttons:

- "First field to search for clip information": A dropdown menu with "Data1" selected.
- "Second field to search for clip information": A dropdown menu with "<not used>" selected.
- "Third field to search for clip information": A dropdown menu with "<not used>" selected.
- "Field to use to for clip information": A dropdown menu with "Data2" selected.
- "Call from identifier": A text field containing "CALL FROM".
- "Area prefix": A text field containing "0".
- "Country prefix": A text field containing "00".
- "Dial prefix": A text field containing "0".
- At the bottom, there are three buttons: "< Previous", "Next >", and "Cancel".

Figure 4-2: configuration window for clip information

4.2.1 Search for Clip information

Up to 3 fields (database fields) can be defined to search for clip information. This can be filled in the first three list boxes.

INI-Setting:

Section: ClipSearch
 Key: DBKey1, DBKey2 and DBKey3
 KeyValue: value between 1 and 12. Value 0 means not used.

4.2.2 Field to use for clip info

In case clip information is found the datafield is copied into the clipinfo (line button), the datafield that should be used can be specified here (value between 0..11)

INI-Setting:

Section: ClipSearch

Key: DBData

KeyValue: value between 0 and 11

4.2.3 Area prefix, Country prefix and Dial prefix

The clipinfo received from the DEFINITY® does not always contain the prefix strings, in order to find the clipinfo anyway three strings can be specified that will be added in the search string thus increasing the chance of finding the clipinfo. The reason for these entries is that the Connecsy database will probably contain these prefix strings in order to use the number for dialling within Connecsy, the DEFINITY® however does not present these strings in the display.

INI-Setting:

Section: ClipSearch

Key: areaPrefix / countryPrefix / dialPrefix

5. DXS/BLF information

5.5 General

DXS/BLF information is also known as busy lamp field indication. DXS stands for Direct eXtension Selection and BLF stands for Busy Lamp Field. Figure 5-1 shows a part of the DXS/BLF window that will be displayed in the Connecsy application (if activated). With the BLF service you can see in the Connecsy application which person/voice terminal is calling or has activated the Send Calls option and which is not. When a red light is showing, the person has actually an off hook situation, which means that the person is calling. A yellow light indicates this person has all its calls sent to an alternative path like voicemail. Also you can double click on one of the fields and the person/voice terminal is direct called (DXS).

The colors Blue and Grey mean that those phones have logged of their extension number and have no coverpath available. (blue is for IP phones and grey is for digital DCP phones)

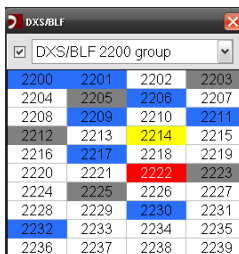


Figure 5-1: DXS/BLF information in the Connecsy application

The attendant console functionality allows us to monitor one hundred lamp indicators at the time. A maximum of 20 hundred groups can be configured in the DEFINITY® and therefore monitored by Connecsy.

Important: The basic DXS/BLF information works only when it is configured in the Definity.

The advanced DXS/BLF (Logged of IP and digital stations) only works when installed the Connecsy Data Service (CDS)

See the chapter 'Definity Settings' on how to configure the DXS/BLF information in the Definity.

The DXS/BLF console can be configured using names instead of numbers in order to facilitate the use of the DXS/BLF console. The console can be viewed in Connecsy in a window using a configurable number of rows and columns. In case the DXS/BLF console is used, a display setting of 1024x768 will improve the way of monitoring the DXS/BLF console since the console can be displayed next to the Connecsy screen. Using 800x600 screen settings the DXS/BLF console will appear when requested and disappears when continuing working Connecsy.

The sheet used for displaying search results from the database can also function as a DXS/BLF console (still monitoring one hundred group at the time). The current (active/selected) hundred group will pass on busy lamp field information to the search result sheet, rows in this sheet that contain extensions within the current hundred group could be marked red indicating busy lamp.

Automatic switching of the DXS/BLF hundred-groups is configurable (in the Connecsy configurator) so that the first row in the sheet containing the search results is examined and depending on the extension the appropriate hundred-group is activated. Therefore when the Connecsy operator is searching for a person and this person is within the e.g. 600 group, Connecsy quickly switches to this hundred group so that busy lamp will indicate if this person is already on the phone. In case the first row is not the desired person the following alternative is in place. For instance the person on the third row is the one that we would like to monitor, it is configurable to automatically switch (if necessary) the hundred-group belonging to this particular person when the row is clicked (still monitoring one hundred group at the time).

5.2 Configure DXS/BLF Information

With the Connecsy Configurator you can configure several settings for the DXS/BLF information. To make these settings you'll need the configuration form displayed in figure 5-2.

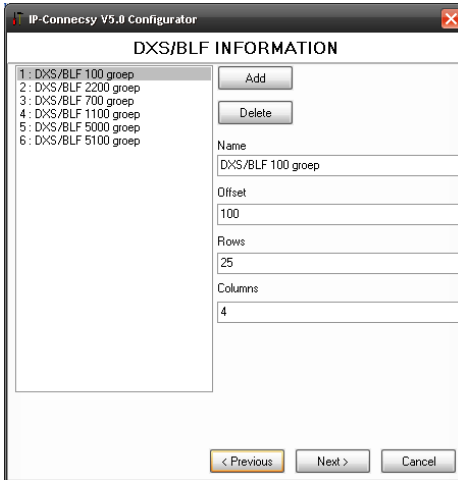


Figure 5-2: configuration form 'DXS/BLF INFORMATION'

5.2.1 DXS/BLF range settings

Maximum 20 hundred groups can be specified (1 - 20) with the values being the offset to the numbers displayed in the DXS/BLF console (500 range will display the numbers 500 - 599). The number of DXS/BLF consoles displayed in the menu is depending on the 'connecsy.ini' file in the language sub-directories. Search for [DSSFormCaptions] inside the language dependent 'connecsy.ini' file. Within that section the entries (0 - 19) can be specified in order to enable or disable the DXS/BLF consoles (in which case the values specified in this section will be discarded).

INI-Setting:

Section: DSSFormRange

Key: The key is numbered dynamically up to 20 range offsets.

KeyValue: The keyvalue is the offset of a defined range.

In this example:

0=0

1=600

2=1000

3=1100

5.2.2 DXS form captions

The DXS form captions are displayed in the listbox. These are the names from the different DXS/BLF ranges. The names are free to define. These settings are user-dependent and language dependent. The settings are set in the 'connecsy.ini' file language directory.

INI-Setting:

Section: dssFormCaptions

Key: number of a range (value between 0..19)

KeyValue: The name of the range

Example:

0=DXS/BLF 00 group

1=DXS/BLF 600 group

2=...

5.2.3 DXS/BLF range (row and column) settings

The values (ranges) specified in the rows/columns text boxes section are used to specify row and column values for a particular DXS/BLF console (see user manual and figure 3). These settings will be made in the 'connecsy.ini' file in the main Connecsy directory.

INI-Setting:

Section:

Key: <hundred number offset>X / <hundred number offset>Y

KeyValue: columns x-axis / rows y-axis

For example:

500x=5

500y=20

600x=5

600y=20

The following options are not configurable with the Connecsy Configurator. These settings must be set with a text editor. These settings are made in the 'connecsy.ini' file in the main Connecsy directory.

5.2.4 DXS/BLF range (number mapping table) settings

This section can be used to map numbers, in case for instance the 500 range has 4 rows (x) and 4 columns (y) and you still want to display 570 use this mapping table.

For example: 515=570

This example means the extension 570 of the DXS/BLF range is mapped to number 515.

INI-Setting:

Section: DSSFormNumbers

Key: In this example the key is 515. This key is dynamic.

KeyValue: This value is also dynamic. This value is the phone number you want map.

5.2.5 DXS/BLF range (name mapping table) settings

Instead of displaying the DXS/BLF numbers you can also display the names. You must assign a name to an extension number.

For example: 570=Helpdesk

INI-Setting:

Section: DSSFormNames

Key: a number of a hundred range where you want to assign a range.

KeyValue: the name, which you want to map to a hundred number.

6. Connecsy Database

6.1 General

In the part dedicated to the database installation we described where the database (sub-directory \DBS\) would be placed during the installation. In the following paragraphs we'll describe the link between the Connecsy application and the database.

6.2 Database interface

The database interface in Connecsy is implemented using a Standard Query Language (SQL). Information from the database is retrieved by dynamically generating SQL queries. In a default situation the Borland Database Engine (BDE) that will provide the link (an alias "Connecsy") with the database handles the SQL queries*.

* Since the database used by Connecsy is of an 'INTRBASE' type the BDE engine can provide the link to the database.

Figure 6-1 displays the link between the Connecsy SQL interface and the database.

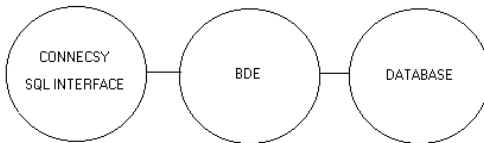


Figure 6-1: link between SQL interface and database

The database type used does not necessarily have to be of type 'INTRBASE': customers may have a database of their own. In case Connecsy has to interface with a database of a different type, it may become necessary to add an additional layer of software between the BDE and the database called the ODBC interface (Open DataBase Connection). (See figure 6-2) For some database types other than Firebird, BDE can provide a direct link to these databases. One such type is Paradox as used in Connecsy V2.x.

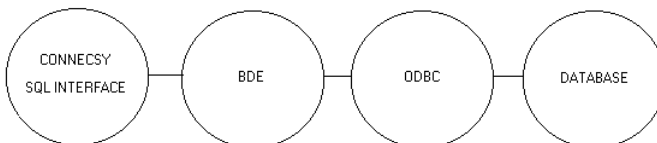


Figure 6-2: use of the ODBC driver

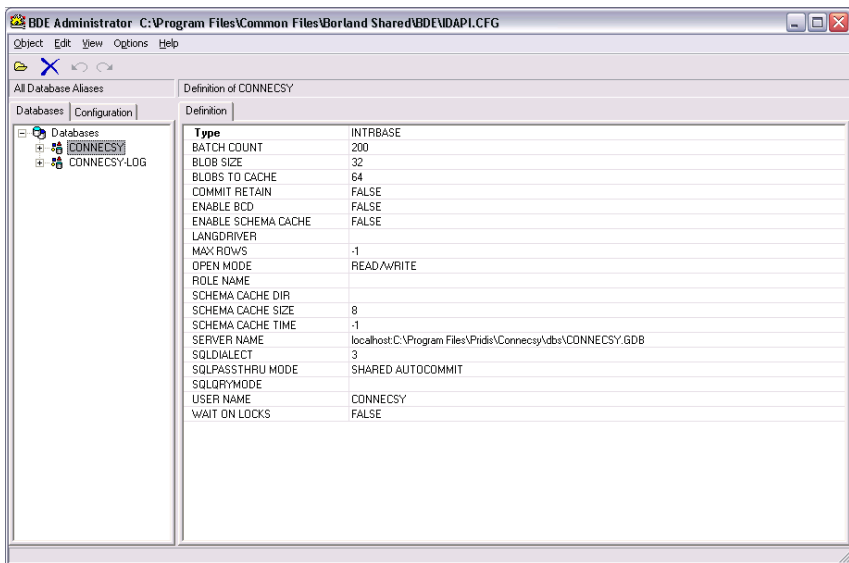
An application can use ODBC data sources. An ODBC driver connection requires
A vendor-supplied ODBC driver
The Microsoft ODBC Driver Manager

The use of the ODBC driver has an impact on the performance of accessing the data. The performance will be lower.

6.3 BDE configuration

The BDE configuration utility is provided with Connecsy and is automatically installed during setup. Start the BDE configuration utility by double clicking BDE administrator icon in the control panel. The BDE configuration utility will display all available database aliases. The setup has already defined a 'CONNESY' database alias, a single click on the 'Connecsy' alias will display the detailed alias settings.

A default Connecsy installation will provide you with the following database alias settings:



By default the database type is of type 'INTRBASE'. The 'INTRBASE' database type does not require additional ODBC installation procedures, which means Connecsy will use a standard SQL interface to access the database and the BDE will provide the link to the database.

A standard Connecsy installation will use a Firebird database table; the location of this database is default set to a local database. However it is possible to use the same database in a network configuration by changing the 'PATH' to a database on a remote database server. The database files will in this case have to be copied (once) to the desired location. The BDE settings on all Connecsy stations have to be adjusted to the recent created network database location.

In some cases the customer has already a database and would like to interface to their own database instead of the Connecsy database. To explain this example for a customer

using an MS Access database, the BDE settings need to be adjusted to meet the customers' requirements. The example MS Access database can be found on the installation CDROM.

The two following steps will change the 'CONNECSY' alias into a 'Microsoft Access Driver (*.mdb)' alias type:

Select and close the 'CONNECSY' alias if necessary (Object menu, Close item),
 Select and remove the current 'CONNECSY' alias (Object menu, Delete item),
 Create a new alias type 'Microsoft Access Driver (*.mdb)' (Object menu, New item),
 Select and rename the new alias to 'CONNECSY' (Object menu, Rename item).

The same actions must be applied when changing de CONNECSY-LOG alias into a 'Microsoft Access Driver (*.mdb)' alias type.

The new 'CONNECSY' alias will contain the following alias settings:

TYPE	Microsoft Access Driver (*.mdb)
BATCH COUNT	200
BLOB SIZE	32
BLOBS TO CACHE	64
DATABASE NAME	CNCSY2DBS
ENABLE BCD	FALSE
ENABLE SCHEMA CACHE	FALSE
LANGDRIVER	
LIST SYNONYMS	NONE
MAX ROWS	-1
ODBC DSN	Connecsy Database
OPEN MODE	READ/WRITE
ROWSET SIZE	20
SCHEMA CACHE DIR	
SCHEMA CACHE SIZE	8
SCHEMA CACHE TIME	-1
SQLPASSTHRU MODE	SHARED AUTOCOMMIT
SQLQRYMODE	LOCAL
USER NAME	

For 'CONNECSY-LOG', the **DATABASE NAME** must be set to **CNCSYLOGDBS** and the **ODBC DSN** must be **Connecsy Log Database**.

6.4 ODBC driver

The ODBC driver is normally a Vendor-supplied driver. The ODBC driver can be installed and configured using the ODBC Administrator (32 bits), which is located in the 'Control Panel'. In some cases the driver must be added to the system, or in the case that the driver is already installed, it will only need some configuration changes. The following two figures display the ODBC administrator screens.

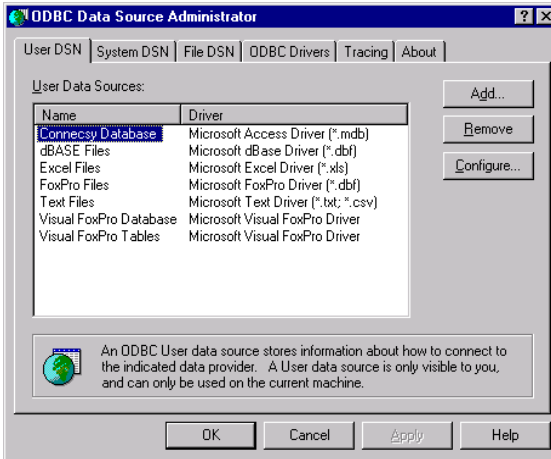


Figure 6-3: ODBC administrator screen

The 'MS Access 97 Database' driver needs to be selected and by clicking the 'configure' button the directory settings of your database can be modified.

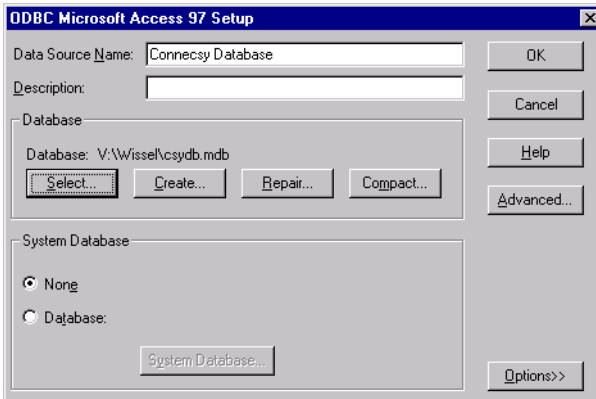


Figure 6-4: ODBC administrator screen (2)

When the ODBC settings are configured properly, you will need to restart Windows™.

6.5 Connecsy Database INI Settings

For the Connecsy database you can also set some settings. These settings are set in the connecsy.ini file in the main Connecsy directory and can only be changed with a text editor. For the Connecsy database you can set several things. These are:

Name of database table,
Database key strings definition,

Database data string definitions (aliases for the key strings used in the configurator).

The database name can be specified. The default name is 'Connecsy'. When you create your own database with another name for the database, you must set the database table name.

INI-Settings:

Section:	DBCommon
Key:	table
KeyValue:	name of the database table

The database key string definitions are the names of the fields in the Connecsy database. These fields are mapped to the DBData section.

For example: DBKeysEntry 1 is mapped to DBDataEntry 0 (DBKeysEntry – 1)

The section of the DBKey definition is DBKeys.

The database string definitions are the aliases of the database fields. These strings are displayed in some of the dropdown listboxes in the Connecsy configurator.

The section of the DBData definition is DBData.

7. The Firebird database engine

From version 5.5, Connecsy can be installed using the Firebird 2.1 database engine as the main database engine. Firebird is an open-source client/server database that's backwards compatible with the Borland Interbase version 6 database engine and offers fast performance with a relative small memory and disk footprint. This database engine is especially well suited for working with large datasets and shared networked databases.

Applications that will be using Firebird to access a database need to have the Firebird client software installed. Machines that will be hosting a Firebird database will need an installed and running Firebird server. In addition to the client and server software, console based administration tools can be installed.

7.1 The client software

The client software can be installed from the Connecsy installation CD-ROM. It does not require any special configuration.

7.2 The server engine

The server engine can be installed from the Connecsy installation CD-ROM. The server engine supplied with Connecsy can be installed on Windows 98, ME, NT 4.0, 2000, XP and 2003 Server. The Firebird server engine is also available for Linux and other UNIX type Operating Systems and can be downloaded from the Internet (<http://www.firebirdsql.org/>). As most users will run the server engine for the purpose of running Connecsy on a Windows based system, this document will only refer to the Windows version of the server engine.

7.3 Controlling the server engine

On Windows 7/vista/XP systems, the engine will be installed as a service so it will be started on boot. You will need to check the Services overview or the firebird server control from the windows control panel, to see if the service is running.



Figure 7-1: Firebird manager window

On all platforms, an applet for controlling the Firebird server called Firebird Manager will be installed inside the Control Panel. Starting this applet will display a window as shown in figure 7-1. With this applet you can control how the server will start (automatic or manually), the location of the root directory for the server and its current running status. Some options may not be available, depending on your Windows platform or running state of the server.

7.4 Configuring the server

The server can be used straight out of the box without the need for additional configuration. If you instruct the installer to install the server engine, the installer will add an extra user named CONNECSY to the Firebird security database with as password connecsy (case dependant). This user is granted the necessary privileges to the database and will be used as the default user by the Connecsy PC attendant application.

If you want to configure the server, you need to install additional tools from the Connecsy installation CD-ROM. Tools included are console based tools. These are meant to be run from the command line and can be used to modify the server configuration; create, modify or remove database; backup and restore databases or to alter security or to run queries on a database.

Every Firebird installation contains an administrative user with the name **SYSDBA**. The default password for this user is **masterkey**. This user will be required when you want to perform administrative tasks on the server, either by the console based tools or the graphical tools. Any user can perform queries as long as they have sufficient rights for the database and the kind of operation they would like to perform.

7.5 Connecting to a database

Firebird supports TCP/IP on all platforms. If you're running an Firebird server on Windows NT 4.0, 2000 or XP you can use NETBEUI, but using TCP/IP is recommended.

7.5.5 Connecting to a database from BDE

To access a Firebird database from BDE, you need to enter a path to the database in the PATH field. If the database is located on the local machine, you can simply enter the file path to the database file. If the database is located on a remote machine, you need to enter the path to the database file on the remote machine. How this path is formatted depends on the transport protocol used. If the database is located on the local machine, but it needs to be accessed by multiple users over the network, it's recommended to use TCP/IP as the transport and to use 127.0.0.1 (or localhost if it's configured in the DNS or local hosts file) for the remote machine.

TCP/IP

If you're using TCP/IP as a transport protocol, the path looks as follows:
<remote machine>:<path to database on remote machine>

For example:

192.168.0.1:C:\program files\pridis\connecsy\dbs\connecsy.gdb

If the remote machine has a name that can be resolved by DNS or a hosts file, you can use this name instead of the IP-address. The path should follow the requirements for paths

for the remote machine, so the above example does apply to a Windows host. If for example the remote machine is running Linux, the path may look as follows:

`linuxhost.mycompany.com:/var/dbs/connecsy.gdb`

NETBEUI

NETBEUI can only be used on Windows NT 4.0, 2000 or XP machines. The path should be formed as an UNC as follows:

`\\remote machine\<path to database on remote machine>`

For example:

`\\DBSERVER\C:\Program files\Pridis\Connecsy\dbs\connecsy.gdb`

8. The SQL Server database engine

In version 5.5, Connecsy provides the support of Microsoft SQL Server or Express Edition as the database engine. However the Firebird 2.1 database engine is still selected as default for the main database engine in the installer. SQL Server Express is a small free database from Microsoft and forms a client/server database and offers fast performance with a relative small memory and disk footprint. This database engine is especially well suited for working with large datasets and shared networked databases.

Applications that will be accessing the Connecsy database need to have the SQL Server Express client software installed. Machines that will be hosting a SQL Server Express database will need an installed and running SQL Server Express server. In addition to the client and server software, console based administration tools can be installed.

8.1 The client software

The client software can be installed from the Connecsy installation CD-ROM by selecting the option SQL Server. Additionally ODBC connections have to be configured.

8.2 The server engine

The server engine can be installed on a dedicated server or on one of the Connecsy PC's. The installation is the same in either case.

Step 1: Download SQL Server 2005 Express Edition

The SQL Server 2005 Express Edition download page has three separate downloads for you to consider.

[SQL Server 2005 Express Edition SP1](#)

Step 2: Identify system requirements

SQL Server 2005 Express Edition does not have significant system overhead requirements, particularly given the overpowered nature of many of today's newest servers. The minimum system requirements for the various SQL Server 2005 versions are:

Table A -- System and software requirements for SQL Server 2005 Express Edition

Feature	Requirement
Version SQL Server 2005 Express Edition	
RAM (minimum)	192MB
RAM (recommended)	512 MB
Drive space	600 MB
Processor (minimum)	600 MHz
Processor (recommended)	1 GHz
Operating systems supported	Windows Server 2003 SP1, Windows Server 2003 Enterprise Edition SP1,

	Windows Server 2003 Datacenter Edition SP1, Windows Server 2003 Web Edition SP1, Windows Small Business Server 2003 Standard Edition SP1, Windows Small Business Server 2003 Premium Edition SP1, Windows XP Professional SP2, Windows XP Home Edition SP2, Windows XP Tablet Edition SP2, Windows XP Media Edition SP2, Windows 2000 Professional Edition SP4, Windows 2000 Server Edition SP4, Windows 2000 Advanced Edition SP4, Windows 2000 Datacenter Server Edition SP4
Software prerequisites	.NET Framework 2.0 & MSXML 6
Other requirements	Server joined to your Active Directory domain.

It is highly recommend to not use one of your existing servers for this purpose. If you don't have any reasonable spare hardware, consider using *VMware Server* or *Virtual Server 2005 R2* and creating a virtual machine. Both products are free. SQL Server 2005 requires the .NET Framework 2.0, which breaks some applications, so keeping the database separate is preferred.

Step 3: Install database software prerequisites

There are a number of software prerequisites for SQL Server 2005 Express Edition. Before you can install the database software, you need to take the necessary steps to get these items installed.

Install the following items in order.

.NET Framework 2.0

Download the .NET Framework 2.0 (x86) from the Microsoft website.

<http://www.microsoft.com/downloads/details.aspx?FamilyID=0856eacb-4362-4b0d-8edd-aab15c5e04f5&DisplayLang=en>

After downloading, execute dotnetfx.exe and follow the instructions to complete the installation.

MSXML6

Download MSXML6 from the Microsoft website:

<http://www.microsoft.com/downloads/details.aspx?familyid=993C0BCF-3BCF-4009-BE21-27E85E1857B1&displaylang=en>.

Execute msxml6.msi. Follow the instructions of this quick installation.

Step 4: Create a SQL Server service account

From a security perspective, it's best to run SQL Server as a normal everyday user. When possible, don't use the built-in service accounts on your SQL Server as this practice does not limit your server's attack surface as much as using an unprivileged user account.

Create a domain account named "SQLEXPRESSUSER" for this purpose. If you're joined to a domain -- which you should be -- use Active Directory Users and Computers. If you're just testing locally, use Computer Management to add the account. Make sure to also assign a strong password to the account.

Also one may consider to configure this user account so that the password does not expire. In addition, anyone logging in as this user -- no one should be doing that though -- will not be able to change the password.

Step 5: Install SQL Server 2005 Express Edition

The sections below describe the installation of SQL Server 2005 Express Edition SP1. The name of the downloaded file is SQLEXPRESS.EXE. Double-click this file to start the product installation. The contents of the file are extracted and the installation begins.

After you start the installation, you have to agree to the typical End User License Agreement after which you're presented with a screen that shows you what prerequisites you need to get the express edition on your system. The installer will handle the installation of these prerequisites for you. Click the Install button to install these items. Once the items are installed, click the Next button.

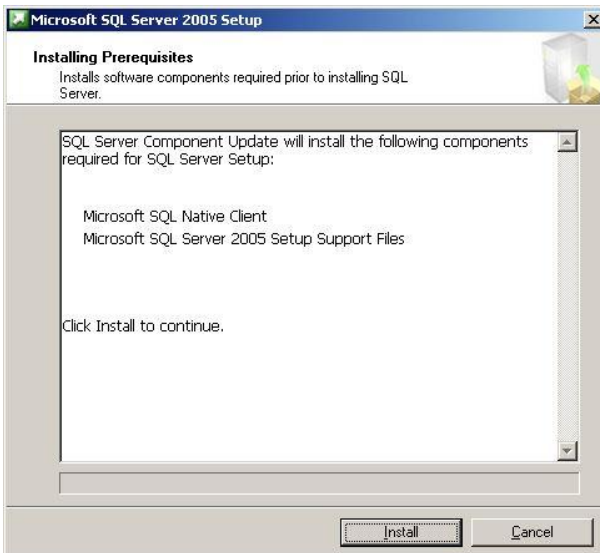


Figure 8.1 The new Native Client and the Setup Support Files need to be installed.

Once the prerequisites are in place the SQL Server installation wizard will start. The first screen you're shown provides you with a list of items that may or may not give you trouble if you continue the installation. In Figure 8.2 below, you can see an example of a test system has a warning regarding the minimum hardware requirements (not enough RAM memory). Note that the installer provides you with a complete status report to make it easy for you to determine what needs to be done. Click Next to continue.

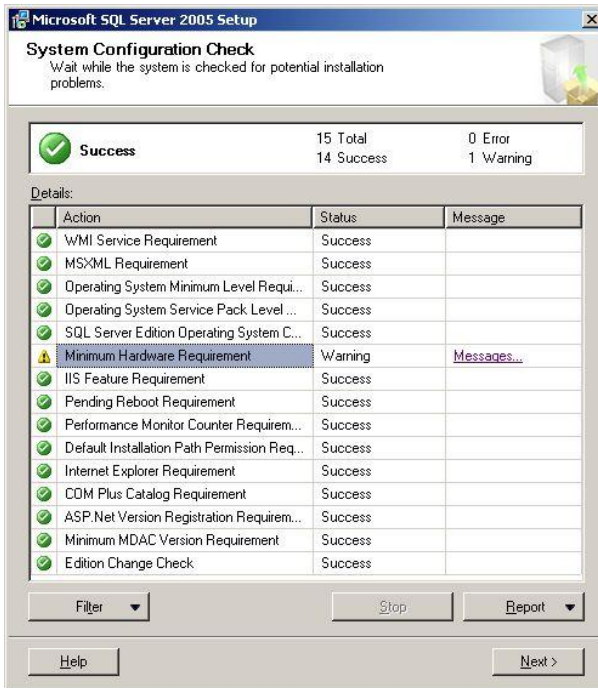


Figure 8.2 If you have any serious problems, correct them before you continue.

After that, you have to provide your name and, optionally, your company's name. You may also uncheck the box marked "Hide advanced configuration options" so you can get a look at these options during the installation.

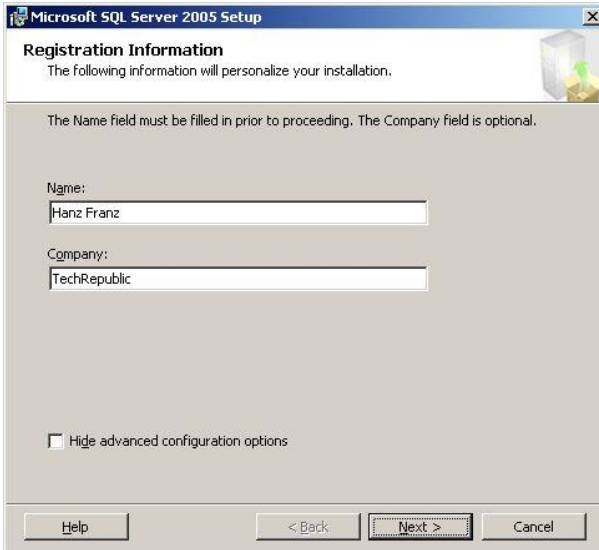


Figure 8.3 Provide registration details for your server installation.

Next, you get to choose which features you want to install. Even though the Express Edition is free, it includes many of the features found in the Standard and Enterprise editions. It is preferred to also mark the Management Studio Express for install. This is an efficient utility provided by Microsoft. It is installed into the default location, C:\Program Files\Microsoft SQL Server. If you want to install to a different location, click the Browse button to choose this new location.

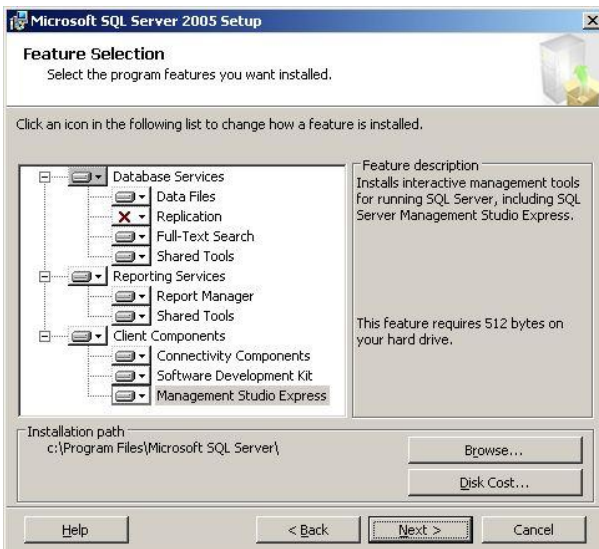


Figure 8.4 Select which features to install

Decide which features you want to install. The only one you absolutely need is the Database Services option, but the others are very useful.

Instances were introduced in SQL Server 2000 and provide a way for you to more seamlessly serve multiple databases from a single SQL Server. SQL Server 2005 Express Edition supports up to 16 named instances while the Enterprise Edition supports up to 50. For this installation, SQL Server 2005 Express Edition is installed as the default database instance on the Connecsy server.

**Figure 8.5** Choose your instance.

Service accounts are used in order to better protect your SQL Server -- and the rest of your network -- in the event of a security breach. If you run your SQL Server under an administrative account, you run the risk of additional damage should your server be compromised. Therefore, run SQL Server 2005 Express Edition under a user account with few, if any, rights.

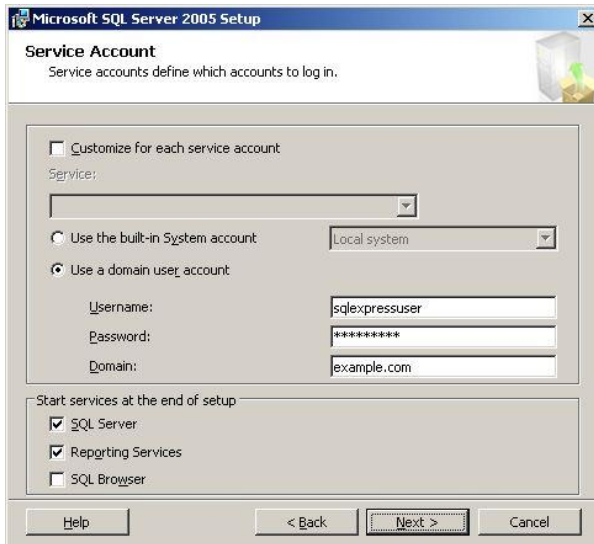


Figure 8.6 Use the account you created before as the service account for SQL Server 2005.

SQL Server can authenticate users in two ways:

- (1) directly against Active Directory (Windows Authentication Mode);
- (2) against its own database (Mixed Mode, since Windows Authentication is also supported).

From a security perspective, Windows Authentication mode is highly preferred. First off, you have only a single user database -- Active Directory -- to worry about.

However, Windows Authentication Mode is not always appropriate. For example, scripting against a Mixed Mode SQL Server works better versus a Windows Authentication Mode SQL Server. If you choose Mixed Mode (which you should) you will also need to provide a password for the SQL Server 'sa' user. Make sure that you assign a strong password if you take this route.



Figure 8.7 Choose the mode under which you want to authenticate SQL Server users.

The next step of the installation involves selecting a collation method, which defines the way that your SQL Server will sort data. If you do not need to worry about backward compatibility with older SQL Server databases, you should choose the Latin1_General collation method, unless, you're using a different language.

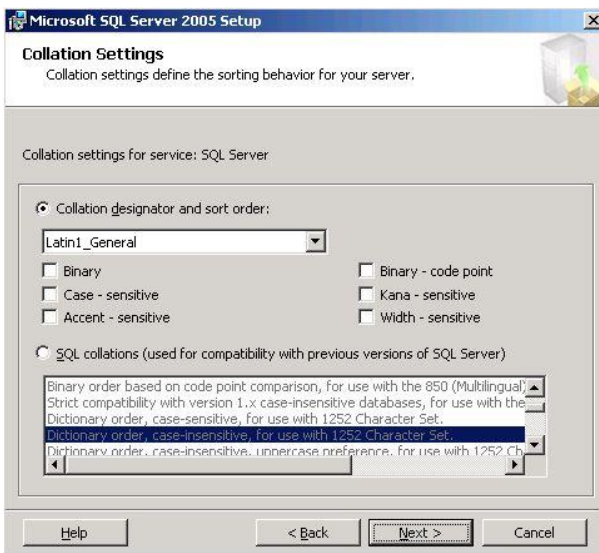


Figure 8.8 Choose your collation method.

SQL Server 2005 Express provides a new feature: The ability to run sub-instances of the product under a normal user account. User instances are useful in situations in which users are logged into Windows under a least-privileged user account. Using user instances, the user can still have SQL system administrator privileges to their SQL "sandbox", but the rest of the system is protected since the user's primary account does not have rights to make global changes. Databases housed under user instances support only a single connection and higher-end features such as replication are not supported.

If you want to support user instances in your installation, make sure the checkbox next to "Enable User Instances" is selected.

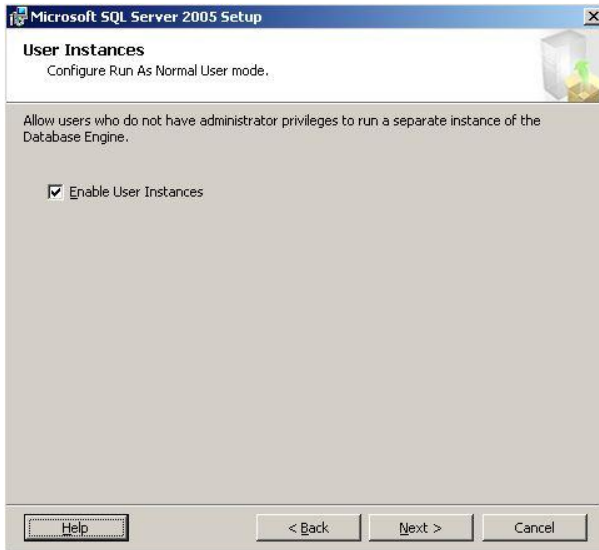


Figure 8.9 Decide whether you want to enable support for user instances.

Microsoft has included an error reporting tool in other applications for quite some time. This feature is now optionally included in SQL Server 2005 Express. Further, Microsoft wants you to send them anonymous usage data about your installation. Keep the default selection where neither option is enabled.

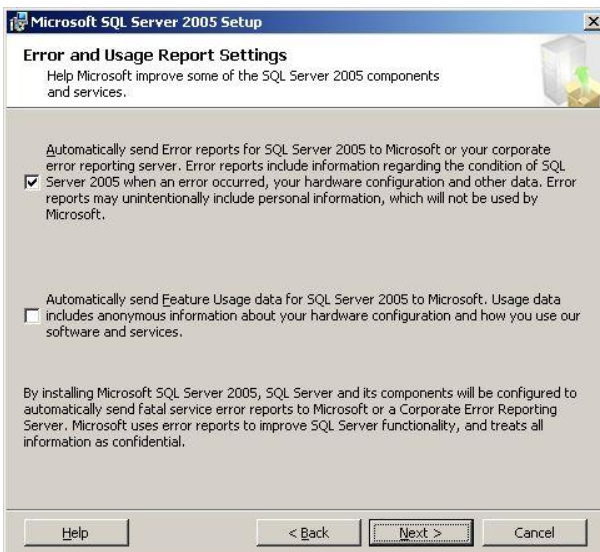


Figure 8.10 Decide what information you want to send to Microsoft.

These are all of your options for SQL Server 2005 Express Edition. Continue on to install the product using these selections.

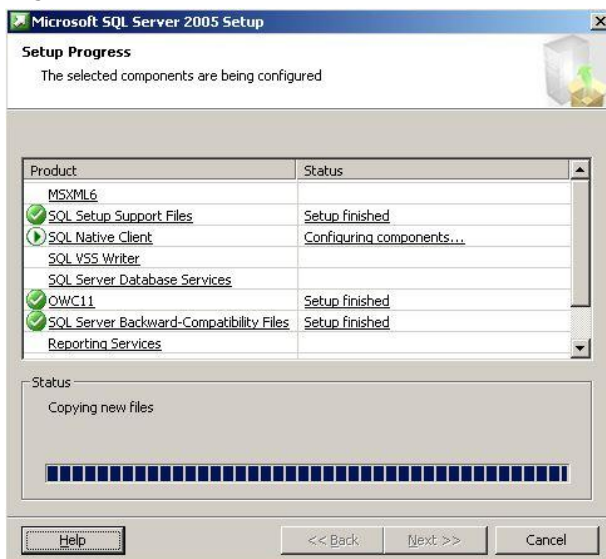


Figure 8.11 The installation's progress.

In some cases, your installation may fail with errors reported for the SQL Native Client and the SQL Server Database Services. In addition, the Workstation Components selection may indicate a failure. If this happens, it may be because you formerly had a conflicting SQL

Server service on the machine and the Native Client previously installed is creating a problem. If this is the case for you, follow these steps:

1. Change to the directory in which you saved the downloaded SQL Server Express 2005 installer.
2. Extract the contents of the installer to a new directory: SQLEXPRESS.EXE
/x:c:\sqltmp.
3. Change to C:\sqltmp\setup.
4. Execute "sqlncli.msi"
5. Choose the "Uninstall" option.
6. Reboot your server.
7. Run the SQL Server Express 2005 installer again. It should be successful this time around.
8. Installed and ready

At this point, SQL Server 2005 Express Edition is installed and can be managed using the SQL Server Management Studio Express tool that was installed along with the database server. To access this tool go to *Start / All Programs / Microsoft SQL Server 2005 / SQL Server Management Studio Express*.

9. Connecsy Import/Export

With the Connecsy V5.5 application you can import and export data. This will be a comma-separated file. You can also import data automatically into the Connecsy database. This chapter explains how the import and export functionalities of Connecsy works. Before you will start with an import or export an administration password is required.

9.1 Connecsy import

The import and export of CSV-files must be started in the Connecsy application. Connecsy appears with a dialog box in which a file can be selected or added.

The import of a CSV-file can be carried out in two ways, namely:

Update the database: only the new records will be added and maybe some will be changed,

Add to the database: all records will be added to the database.

When you start the import, Connecsy will let you make a choice between these two ways.

There are three types of files, which the Connecsy application can import. These are the following types:

Comma Separated (CSV)

Definity ® file (*.LST)

9.2 Comma Separated Value (CSV-file)

A CSV-file is a file where the data fields are separated with a comma. The data is placed between double quotes. In the following example a comma-separated file is displayed.

Example:

```
"0","1","2","3","4","5","6","7","8","9","10","11"
"13","Jansen","Manager"," "," "," "," "," "," "," "," "
"14","Gerritsen","Programmer"," "," "," "," "," "," "," "," "
"15","Pietersen"," "," "," "," "," "," "," "," "," "," "
```

The first line defines the field numbers where the data will be placed in the Connecsy database. This means that the name 'jansen' will be placed in field number 2. When the numbers 1 and 2 are switched the name 'jansen' is placed into field number 2.

Important: Connecsy has 12 database fields. In the CSV-file you must number the header 0 to 11 & the 12th field must be named: note.

9.3 Connecsy Export

For the export there is one type of file that the Connecsy application can export. This is the following type:



Comma Separated (*.CSV)

When the data is exported to a comma separated file you must first specify a location to where the data will be exported. When you have selected a location a dialog window will appear inside which you can toggle the fields on or off for export. These values will be set in the 'connecsy.ini' file in the main Connecsy directory. You can also export the ID field. What the purpose is of this ID will be described in paragraph 'Comma separated file'.

INI-Setting:

Section: ExportCSV
Key: CheckBox0, ... CheckBox11 / CheckBoxId / CheckBoxKey
KeyValue: 0: no export / 1: export

Example:

1,1,1,1,0,1,1,1,1,1,0,0

9.4 Connecsy Automatic Import

When you want frequent to update your local database with the central database, you can use the Connecsy automatic import. The settings of the automatic import of a CSV-file in Connecsy are also configured in the 'connecsy.ini' file in the main Connecsy directory. These settings are made in the Connecsy application and not in the Connecsy Configurator. In this paragraph all the settings of the automatic import will be explained with their link to the 'connecsy.ini' in the main Connecsy directory. The automatic import pop-up screen is displayed in figure 8-1.

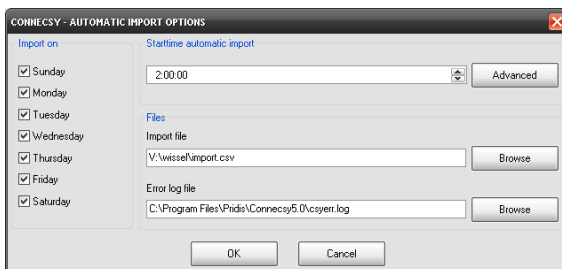


Figure 9-1: Connecsy automatic import option window

9.4.1 Day of import

The day of automatic import must be specified.

INI-Setting

Section: AutoImport
Key: DaysToImport
KeyValue: The automatic import process should run on the following days: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday and Saturday. These days are displayed in the INI-file displayed as 7 numbers separated by a comma. The values of these numbers are 0 or 1. 0 means no automatic import on that day and 1 vice versa.
Example:

DaysToImport = 0,0,0,1,0,0,0

9.4.2 Starttime of automatic import

The following entry specifies the time stamp to start the import process.

INI-Setting

Section:	AutoImport
Key:	TimeToImportHour TimeToImportMin TimeToImportSec
KeyValue:	the value of the time units described above

9.4.3 Import and error log file.

These text boxes specify the file, which must be imported, and the name of the file in which the log data will be added. The following entries will be specified:

INI-Setting

Section:	AutoImport
Key:	ImportFileName ErrorLogFileName
KeyValue:	the file names must be specified with full path name and driveletter.

9.4.4 Connecsy Automatic Import (Advanced)

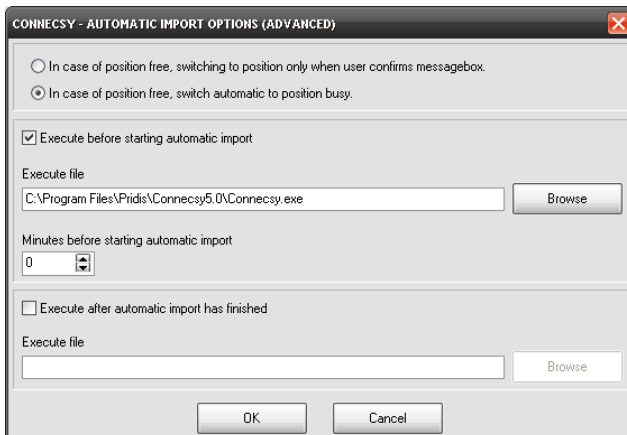


Figure 9-2: Connecsy automatic import advanced options window

With the automatic import advanced window (see figure 8-2) you can specify which files must be executed before and after the import. An example is retrieving a CSV-file by an ftp-session. When the CSV-file stays on an ftp-server you'll need this option. First you must open the ftp-client and download the file from the ftp server. After this the file will be imported. The time between it is set with the parameter 'Minutes before starting automatic import'.

The time between it is longer then the specified time (about 5 minutes). In this time the Connecsy application is polling to the CSV-file to look at the file size. When this is not increased during the 5 minutes the application wait for the time you have specified and then start with the import.

After the import you can for example send a mail to somebody so he/she knows that the file has been imported. For whom this mail is aimed for you must activate them with the checkbox.

INI-Setting

Section: AutoImport
Key: DoPreImportRun/DoPostImportRun (activate the options)
KeyValue: YES / NO

Key: PreImportFileName/PostImportFileName (file names to execute)
KeyValue: the executing filename with full path and drive letter

Key: PreMinuteWait
KeyValue:

9.5 Comma Separate Value (CSV) with ID

Connecsy make the distinction between imported records and records added in the Connecsy database via the edit fields in the Connecsy application. These are manual made records. After importing records Connecsy will remove the previous records. The manual made records will however never be removed. It is possible to import data on a special manner so that Connecsy sees the data as manual made data. Connecsy can make this distinction by the record ID.

With Connecsy application a export file can be made to include the record ID. The record number for a 16 bit Connecsy version of Connecsy is 32 bit (4 bytes). The highest "nibble" of a manual-imported record is always zero. In a decimal notation you cannot see the difference directly.

Example of an export file including a record number:

```
Id,"0","1","2","3","4","5","6","7","8","9","10","11"
107826796,"1","manual imported"," "," "," "," "," "," "," "," "," "," "," "
913133217,"1","imported record"," "," "," "," "," "," "," "," "," "," "," "
913133218,"1","staff"," "," "," "," "," "," "," "," "," "," "," "
```

When the record ID will be converted to a hexadecimal notation you can see that the highest number (nibble) is zero for hand-imported files.

	Decimal	Hexadecimal
Hand imported	107826796	066D4E6C
Imported files	913133217	366D4EA1

Important: By first exporting all records (including the record ID) and after that muting the record number, the Connecsy application will recognize this imported file as a manual imported file. These records will not be deleted with regular import actions.

With a spreadsheet the record ID can be muted. Therefore you must use the following function:

```
Function NewID(id as long) As Long
    Const HEX_0FFFFFFF As Long = 268435455
    NewID = id And HEX_0FFFFFFF
End Function
```

The first line (see CSV-example above) defines the fields (places) of the data in the Connecsy database. This means in this example that the data 'manual imported', 'imported record' and 'staff' data will be placed in at field 2.

9.6 Import / Download Definity® file

When a Definity® file must be imported, the Connecsy application makes a serial connection with the Definity. The Connecsy application collects only the extension and name (person) data from the Definity®.

The Definity need commands if you want to collect the necessary data. When the data will be collected, the Definity® will check the start and the end string. In case they are not presented, an error is reported and the import process is aborted. Setting the text to the 'default' string means that default no checking will be done for strings at the beginning and the end of the file.

```
INI-Setting (key and KeyValue)
StartString=STATIONS
EndString=successfully completed
```

Before collecting the Definity needs some commands. We describe now the key's of the INI-files with the command's:

INI-Setting (key and KeyValue)	
ListString = list station	get a list of all stations
LogoffString = logoff	logoff on the Definity®
SuccessString=successfully completed	
LoginString=Login:<login name>	the login name of the Definity®
PasswordString=Password:<password>	the password of the Definity®
IncorrectString=INCORRECT LOGIN	incorrect account
CommandString=Command:	
TerminalString=513	
PageString=NEXT PAGE	

To get the Definity data into the Connecsy database you must specify the parameters related to the position of the data in the Definity file. The data in the Definity file is placed in fixed places.

For the extension we need the position, length and destination of the extension data in the Definity® file. This is also for the name data.

INI-Setting (key and KeyValue Extension)

Ext=0 (position in the Connecsy database)

ExtPos=1

ExtLenth=27

INI-Setting (Key and KeyValue Name)

Name=1

Definity® till version 6

NamePos=23

NameLength = 15

Definity® from version 6

NamePos=15

NameLength=27

10. Connecsy Sounds

10.1 Sound types

Connecsy is able to distinguish between 5 different types of calls from the Definity®. The Definity® pass the type of an incoming call to the Connecsy application so the Connecsy application knows what type of call it is. The Connecsy application maps the different call types to Windows sounds. Combined with a visual indication the Connecsy user can hear and see what type of call it is.

Previous Connecsy versions mapped the sounds to Windows system events. From Connecsy version 4.0 and up, a separate sound group can be found for Connecsy inside the Sounds control applet (see picture 9-1).



Figure 10-1: Connecsy sound group inside the Sound control panel applet.

This group will enable the user to differentiate between Connecsy sounds and system sounds more easily. If no specific Connecsy sounds are defined, Connecsy will redirect the sound to the system sound as described in the following table:

Definity Sound	Connected Windows Sound	Switched by
Incoming call ringing	Exclamation	Programmed feature button in-ringoff
Held call ringing	Question	Programmed feature button re-ringoff

Waiting call ringing	Question	Programmed feature button cw-ringoff
Unkown call ringing	Exclamation	
Emergency call ringing	Asterix	

The Definity sounds are also displayed in the upper left corner of the Connecsy screen. In case a soundcard does not work properly, the Connecsy user can see there is a call and what type (see picture 9-2).

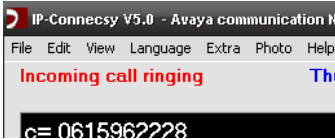


Figure 10-1: display of the Connecsy sound types

In the following paragraphs we will explain about some sound types and some settings that must be made.

10.2 Reminder on Hold Timer for "held call ringing"

When a call is placed on hold, a timed reminder will automatically alert the attendant after an administrated time interval ("held call ringing"). The attendant must re-enter the call before they can handle another call. To handle more than one call, the interval must be not too short (normal 30 sec). Change the time interval by typing "change console" at the TERRANOVA command prompt and change the parameter of the field "Time Remainder on Hold (Sec):" (How you can make this setting will be explained in the Definity Settings).

10.3 Waiting call ringing

When the attendant is handling a call, a second call is placed in the attendant queue. The "Waiting call ringing" alerts the attendant when a call is placed in queue. When the attendant is called with the attendant extension, the call is then placed in the queue and the attendant is not alerted by a Definity sound! For this reason Sound Queue Ringing is added to the Connecsy application, so to disable "Waiting call ringing" you have to do it in two places:

Deactivate the cw-ringoff feature button in the Connecsy application and Uncheck "sound (Queue ringing)" in the menu item "extra" of the Connecsy application.

10.4 Sound Queue ringing

When there is a waiting call the Definity gives two messages to the Connecsy application: "Waiting call ringing"
"Queue" (this means there is a call in the queue)

Sound queue ringing is a feature of Connecsy. This sound alerts the attendant when a call is placed in the attendant queue. When the attendant queue starts, the word "Queue" is

displayed in the middle of the Connecsy screen and the queue ringing sound is heard (if enabled). This sound will be fired once.

The sound of the queue ringing can be switched via "Sound (Queue ringing)" in the menu item "Extra" of the Connecsy application. You can toggle this option on and off.

When there is a waiting call the Definity gives two messages for the call: the Definity sound "waiting call ringing" and the "queue". The Connecsy application uses the queue sound as leading. When the queue sound is toggled off the sound of the "waiting call ringing" is used. When all of the sound must be disabled, the feature button "cw-ringoff" (in the Connecsy application) and the option "Sound (Queue ringing)" must be disabled (also explained earlier in this paragraph).

Important: When the feature button "cw-ringoff" is disabled there will appear no "waiting call ringing" string in the upper left corner of the Connecsy screen.

This means for the waiting calls you have three options of sound. These are the following:

None

Queue ringing

Waiting call ringing

11. Connecsy Queue information

11.1 Configuring Queue Information

In the Connecsy application you can display queue information. Up to 3 queues can be monitored. Before this function is activated, the Connecsy configurator must configure the application. These settings can be configured with the window 'queue information'. All the settings of queue information are located in the connecsy.ini in the main Connecsy directory. The queue information can be configured with the Connecsy Configurator. The screen you will need is displayed in figure 10-1.

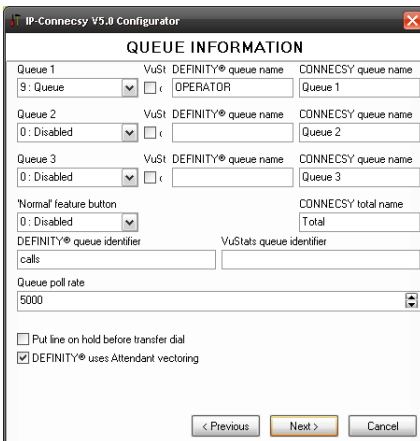


Figure 11-1: configuration window for queue information

For each attendant station some feature buttons must be programmed to retrieve the queue status information.

11.1.1 Definity Queue Types

Two types of queue calls can be configured. These are:
atd-qcalls and
q-calls

These must be programmed in the Definity® by the features. See the chapter Definity Settings. In the first left column of combo boxes you can select the feature button where the queue information function is programmed.

INI-Setting

Section: QueueStatusOption
Key: QueueButtonCode1/ QueueButtonCode2/ QueueButtonCode3
KeyValue: The value of the feature button (0: disabled / value 1..24)

11.1.2 Definity Queue names

These names are used to identify a queue when the display message is received and interpreted by Connecsy to extract the queue-information.

For q-calls use hunt group names:

QueueName1 = ACD 1 (31)

QueueName2 = ACD 2 (32)

QueueName3 = ACD 3 (33)

For atd-calls use the name QueueName3 = OPERATOR

These names must be the same as the name set in the Definity. You have to check by the console parameters and look at the 'Attendant Group Name'.

When the queue information is displayed in the Connecsy screen you can give a label for the queue fields. These are the Connecsy Queue names. There is also a field with the total calls in the defined queues. The label of this field can be set. The labels of these fields are set in the connecsy.ini in the language directory of Connecsy.

INI-Setting

Section: DisplayCaptions

Key: q1, q2, q3, total

KeyValue: the names of the queues.

11.1.3 Definity Queue String Identifier

The queue string identifier is the string preceding the quantity of calls waiting in the queue. This string is depending on the language setting in the DEFINITY®. This string will be placed in the box for clip information.

'ACD 1 (31) Q-time 0:00 calls 0' is received as a queue info display message where the queue identifier is the string 'calls'.

'OPERATOR Q-time 0:00 calls 0

Connecsy matches the words defined at the Definity Queue label with the Definity Queue String. When the identifier words match, Connecsy extracts the queue information.

The word 'calls' in the displayed string above is configured in the connecsy.ini file.

INI-Setting

Section: QueueStatusOption

Key: DefinityQueueIdentifier

KeyValue: calls

11.1.4 Connecsy Poll-rate

The poll-rate is the rate with which Connecsy requests the queue-status information in case of an active (calls waiting) queue. This must be defined in milliseconds with a minimum of 1000.

INI-Setting

Section: QueueStatusOption



Key: QueuePollRate
KeyValue: defined in milliseconds with a minimum of 1000

11.1.5 Put line on hold before transfer dial

In case you prefer to transfer calls by directly typing the phone number of the person you want to call, there is a possibility to put the line on hold before you start typing. When you turn this checkbox on, the caller will be set on hold when the operator opens a call dialog window.

11.1.6 DEFINITY® uses attendant vectoring

If Attendant Vectoring is enabled on DEFINITY® you can check this checkbox. This prevents problems with the display of the queue information in Connecsy.

12. Presence Data Service

12.1 General

The Connecsy Data Service fetches status and presence information from one or more Communication Manager systems via Snmp and stores this information into the Connecsy database. Each of these connections should be configured.

12.2 Single Attendant Setup

For a single attendant configuration the Data Service will be setup according to the following figure:

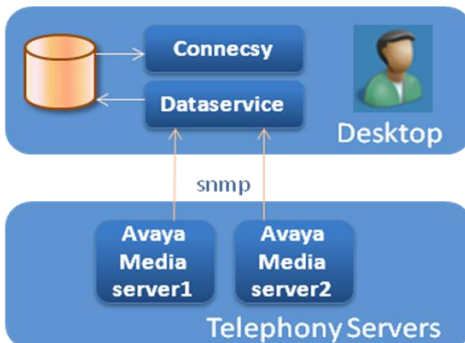


Figure 12-1: Dataservice setup with a single attendant

The configuration is stored and changed directly in the Dataservice config file. See chapter 1 for details.

The configuration of the SNMP connection with the Avaya Communication Manager servers is described in chapter 15.50.

12.3 Automatically add extensions to Connecsy

The Dataservice fetches its status information from each extension within Communication Manager. Normally only states of extensions contained within the Connecsy phonebook database are passed, but one may select to have the status of any extension added automatically to the Connecsy phonebook database. However keep in mind that the other organisation information like department, person function, etc. should be added afterwards.

AddRecordIntoDBIfExists may have value true or false. When true, then the states of all CM extensions are updated in the Connecsy database (the phonebook) regardless of whether they were in the guide. When this value is set to false, then he only updates the status of the extensions that are in the Connecsy database.

```
<appSettings>
```

```
    <add key="AddRecordIntoDBIfExists" value="false" />
```

```
</appSettings>
```

13. Connecsy Configurator

All Connecsy configuration settings are stored in ('connecsy.ini') files, using sections and entries. Depending on the type of settings, the location of the ini file is different. Currently Connecsy is using three types of 'connecsy.ini' files: language-dependent, language-independent and user-dependent files. The language-dependent and independent files are created the first time the application is started. The 'connecsy.ini' files created at this point are copied from the 'connecsy.default' files. The reason for not creating the 'connecsy.ini' file during the setup procedure is to prevent Connecsy stations that are upgraded to a newer Connecsy version to lose their customised configuration settings. After installing Connecsy using the setup procedure, customisation can be done using an additional setup tool ('ConnecsyConfigurator.exe'). The next chapters will refer to this setup tool using figures for those of the section and entries incorporated in this tool.

13.1 Connecsy language selection

Connecsy language selection:



When running the setup tool, settings can be changed for all languages, switch languages using the language-setting drop down list. The language displayed in the drop down list is the current language, which is set in the language menu item in the Connecsy application (see figure 13-1).



Figure 13-1: language selection window

Language-dependent files contain all text that is used for the user-interface. Up to 8 languages can be defined, by default three languages are enabled.

Dutch	(nld sub-directory)
English	(enu sub-directory)
German	(deu sub-directory)

Other languages:

Spanish (esp sub-directory)
French (fra sub-directory)
Italian (ita sub-directory)
Unknown (xxx sub-directory)
Unknown (yyy sub-directory)

13.2 Version codes

Connecsy V5.5 is protected by a hardware dongle. Without this dongle, Connecsy refuses to start-up normally and to connect to the PassageWay. Connecsy can still be run in demo mode or data management mode though.

The dongle also stores license information for Connecsy. License information contains the serial number, installed modules etc. To dynamically alter your license without the need of replacing dongles, the Configurator uses version codes to program your license into the dongle. The version codes are hooked tightly with the dongle you received and the serial number and can't be used with another dongle. Your Connecsy package contains a license information card that holds the license information and the version codes. If you make modifications on your license – like purchasing extra modules – your vendor will send you a new set of version codes.



Figure 13-2: Version Codes display

The Version Codes display (see figure 11-2) in the Configurator is used to update the license information inside the dongle. Enter the three version codes inside the three fields and click **Register**. If the entered codes resemble a valid license, a window with the new license information will be shown. Click **Yes** to accept the license and program it into the dongle or press **No** to retain the current license.

13.3 Common Settings

With this display you can configure the most common used settings of the Connecsy application. All these settings are set in the 'connecsy.ini' file in the main Connecsy directory. Figure 11-3 displays the configuration form to configure the 'common' settings.

COMMON	
Title (added to the program title)	Avaya communication Nederland B.V.
Automatic clear (after pressing escape and release)	TRUE
Cursor position (after pressing escape)	Ext
Automatic sort (after search results are displayed)	1 : Ext
Display alphanumeric keyboard (after pressing edit buttons)	FALSE
DXS Simulation (after search)	press DXS/BLF hundred group button of selected row and change form (also on click)

Figure 13-3: configuration window 'common' settings

13.5.0 Title

With this parameter you can give a title to the Connecsy application. The application title is extended with a custom string specified with the title entry.

For example: "IP-Connecsy V5.5 – Avaya Communication Nederland B.V."

INI-Setting:

Section: Common

Key: Title

KeyValue: Avaya communication Nederland B.V.

13.3.2 Automatic Clear

This option provides an automatic clear of the search fields. After 'CANCEL' and 'RELEASE' the escape button will be automatically pressed. The search fields and the datasheet will be cleared. The cursor jumps into its start position set at the cursor position.

INI-Setting:

Section: Common

Key: AutoEscape

KeyValue: Value 1 means automatically pressed (value between 0 - 1)

13.3.3 Cursor position

When the escape button is pressed, the edit fields are automatically cleared and the cursor position is located in the field you have selected with this option. In the configurator you will see names, but the INI-file works with numbers. The fields are numbered of 0 to 11.

INI-Setting:

Section: Common

Key: DefaultEscPosition

KeyValue: 0 (value between 0 - 11/ in configurator screen 'extension' to '[field12]')

13.3.4 Automatic sort

The 'AutoSort' parameter enables automatic sorting after searching the database, the number is indicating the column that should be sorted.

INI-Setting:

Section: Common

Key: AutoSort

KeyValue: value between 0 - (maxcolumn-1)

13.3.5 Display Alphanumeric keyboard

In case a touch-screen monitor is installed it could be useful to enable the following parameter (value TRUE or FALSE). When indicating TRUE the buttons next to the edit fields on the Connecsy main screen will cause the alphanumeric display to appear, otherwise the buttons will start a search action on the search-string present in the particular edit field.

INI-Setting:

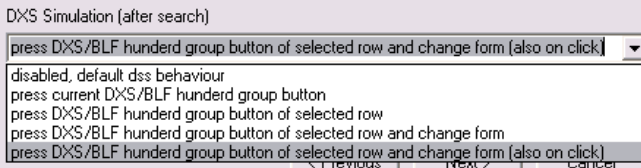
Section: TouchScreen

Key: keyboard

KeyValue: True or False

13.3.6 DXS Simulation

The purpose of this setting is explained in by the DXS/BLF information. With this option you can set an action after a search. You can automatically press a DSS button. The values that you can set are given in the following drawing.



INI-Setting:

Section: Common

Key: autoDSSBtn

KeyValue:0: disabled, default dss behaviour

1: press dss button of current dss hundred-group

2: press dss button of selected row dss hundred-group

3: press dss button of selected row dss hundred-group and change

dssForm (only after search)

4: press dss button of selected row dss hundred-group and change dssForm (after search and on a row click)

Important: It's recommended to select option number 4.

13.4 Communication Interface

The communication interface window (figure 12-4) of the Connecsy Configurator provides the communication settings for the communications links to the Definity® and the serial settings needed to import data from the Definity® into the Connecsy database.

Figure 13-4: configuration window 'communication interface'

13.4.1 Communications link to DEFINITY®

Connecsy 5.5 and up connects to the Definity® PBX and any other supported Avaya PBX in the future by external modules called Media Connectors. Connecsy 5.5 comes installed with two types of Media Connectors by default:

- Serial DCP. This will connect Connecsy to the Definity® via a serial connection and a PassageWay® adapter. This media connector is provided for legacy purposes.

H.323 IP-Console protocol. This will connect Connecsy to the Definity® over an IP-network.

The 'Media Connector type' combo box enabled you to select a specific media connector. Two media connectors are provided by default – Serial DCP and H.323 IP-Console protocol – and their specific configuration details will be described in this document. Other types may be installed and listed as well, but configuration details will not be included in this document. To configure a media connector, select the desired type and click the 'Configure' button. This will pop up another dialog that enabled you to change the configuration details specific for the selected media connector type.

13.4.1.1 Serial DCP Media Connector

The configuration dialog for the Serial DCP Media Connector enables you to specify the communications port to which the PassageWay® adapter has been connected and the baud rate to use.

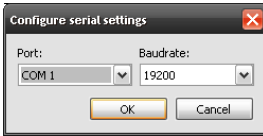


Figure 13-5: Serial DCP configuration dialog

13.4.1.2 H.323 IP-Console Protocol Media Connector

The configuration dialog for the H.323 IP-Console Protocol Media Connector enables you to specify the settings to enable Connecsy to communicate with the Definity® over TCP/IP.

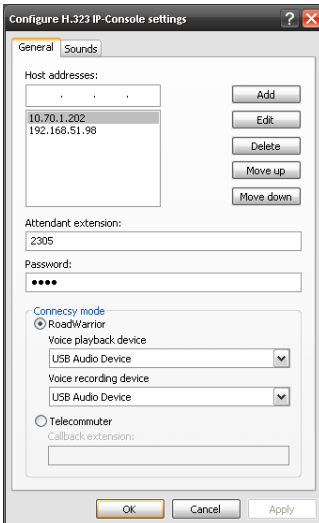


Figure 13-6: H.323 IP-Console Protocol configuration dialog, General page

Host address

Enter the IP-address or DNS-listed host name of the Definity® to which to connect.

Attendant extension

Enter the extension of the attendant console position you want to use. Only one console can be active at an attendant console position at any given time.

Password

Extensions configured on the Definity® for use over TCP/IP require a password. Enter the password associated with the 'Attendant extension' here.

Callback extension

This release of the H.323 IP-Console Protocol Media Connector uses Telecommuter mode only. Telecommuter mode requires a separate analogue, digital or IP-device to provide the voice-path. Enter the extension of the device that should provide the voice-path here. This can also be any valid external phone numbers, e.g. cellular phones.

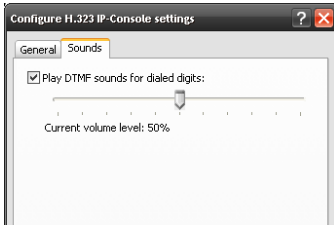


Figure 13-7: H.323 IP-Console Protocol configuration dialog, Sounds page

Play DTMF sounds for dialed digits

In Telecommuter mode, there's no direct path between the Connecsy console and the device that provided the voice-path. This also means that when dialling digits, there are no DTMF sounds played over the voice-path. When checked, Connecsy will play DTMF digits over the default sound device of the PC being used.

13.4.2 DEFINITY® database import (Port and Baud rate)

Serial communication via a terminal port of DEFINITY® is only required in the case that the download of the stations configured in the DEFINITY® is done via an import process. Connecsy will establish communication with DEFINITY® and will gather the station data and write this to a file. This file is imported when the download is completed. For this communication the port number and Baud rate must also be specified.

INI-Setting:
 Section: Communication
 Key: comDef and baudRateDEF

13.5 Database

This window enables you to change the database table used to retrieve/store data from/in or to enter user and password information if you're database requires a login (see figure 13-8). The password entered will be stored in an encrypted format.

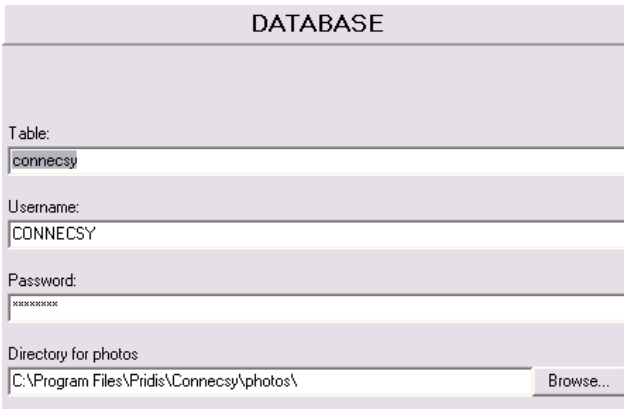


Table:

connecsy

Username:

CONNESY

Password:

XXXXXXXX

Directory for photos

C:\Program Files\Pridis\Connecsy\photos\

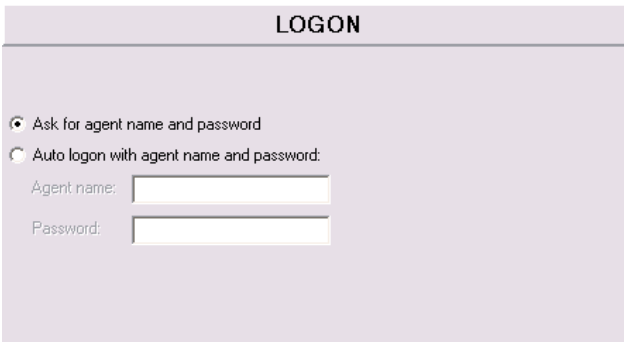
Browse...

Figure 13-8: configuration window for 'Database'

This window also enables you to specify the directory used to store images of the people stored inside your database.

13.6 Logon

This window allows you to select between auto-logon and manual logon to Connecsy. When manual logon has been selected, Connecsy will ask for an agent name and password. When auto-logon has been selected, Connecsy will logon using the agent name and password provided inside this window.



LOGON

☒ Ask for agent name and password

☐ Auto logon with agent name and password:

Agent name:

Password:

Figure 13-9: configuration window for 'Logon'

13.7 Feature buttons

With the configuration window 'Feature Buttons' you can configure the feature buttons. This window is displayed in figure 12-10.



Figure 13-10: configuration window 'Feature Buttons'

The feature buttons defined in the "connecsy.ini" file should match the settings in the DEFINITY®. The names of the feature buttons in Connecsy are only used as a reminder for the user (user-dependent names), in case another feature button is programmed in the DEFINITY®, unpredictable results will occur.

When the configurator is started for the first time the settings will be loaded from the connecsy.default from the language directory.

INI-Setting:(Feature captions)

Section: featureCaptions
 Key: Number of the feature button (value 0..23)
 KeyValue: Caption of the button

INI-Setting: (For the hints the same principle)

Section: featureHints
 Key: Number of the feature button (value 0..23)
 KeyValue: Hint of the button

Example:

```
[featureCaptions]
0=Split
4=Priority
5=Hold
```

```
[featureHints]
0=Split
4=Priority Dial
5=Hold
```

13.8 Function buttons

With the configuration window 'Function buttons' you can configure the settings of the function buttons. This configuration window is displayed in figure 11-11. The function of each button is fixed. There are several buttons, which can be programmed. The captions and hints of the function buttons are configurable. This is the same for the bitmaps of the function buttons in the Connecsy application. These settings are configurable the 'function button' display of the Connecsy Configurator. Some function buttons are connected/linked to feature buttons. This will be explained later.

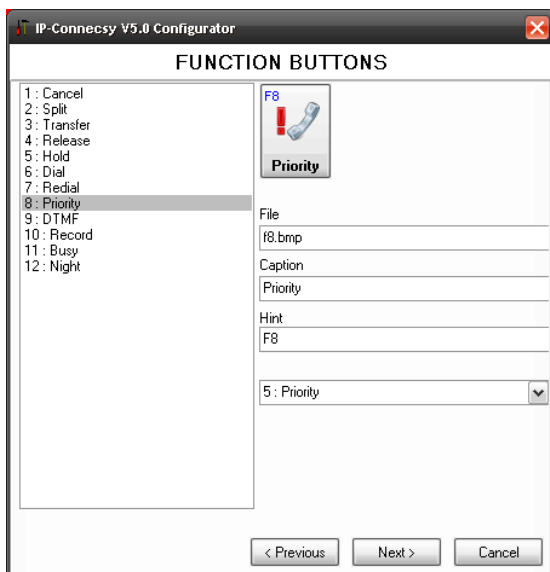


Figure 13-11: configuration window 'Function Button'

13.8.1 Caption and Hints of the function keys

The caption and hints are set in the same manner as the feature buttons. The function buttons have two more settings to configure as the feature buttons. These are (picture-)file and purpose. With the file setting you can specify a picture you will see in the function button. The captions and hints are set in the 'connecsy.ini'-file in the language directory.

13.8.2 Functions key bitmaps (file)

The bitmaps can be changed with this parameter. The following entries can be used to specify bitmaps that are displayed in the function keys f1 (0) till f12 (11). The entries should only contain bitmap names (no directories allowed), by default the bitmaps will be loaded from the sub-directory (bmps) in the Connecsy main directory.

INI-Setting

Section: buttonBitmaps

Key: button value (f1(0) .. f12(11))

KeyValue: filename of the bitmap

13.8.3 Programmable function buttons

Some of the feature buttons are default connected with the function buttons of the Connecsy application, if those buttons are not programmed in the DEFINITY®, the specified function buttons in the Connecsy application will not work.

F2 is connected with feature button 1: Split.(fixed)

F7 is connected with feature button 10: last-numb (Redial)

F8 is connected with feature button 5: priority

F11 is connected with feature button 24: pos-busy

F12 is connected with feature button 23: night-serv

This can be changed by the combo box purpose. The function buttons just described are the default settings.

Important: Program the Definity® feature buttons in the same way.

In the INI-file the function key (button) F7 is the redial button, the value is indicating under which feature button the redial function is located (value between 1 - 24). This is the same for the function key (button) F8, which is the priority dial button. These buttons are connected/linked to a feature button. This can be selected with the parameter 'purpose'.

INI-Setting

Section: common

Key: RedialButton / PriorityButton

KeyValue: the value of the feature button under which the redial function is located

Function keys (buttons) F10, F11, F12 are also programmable. This means: you can connect it with a feature button.

INI-Setting

Section: Common

Key: F10Button / F11Button / F12Button

KeyValue: the value of the feature button under which a function is located.

F10, F11: 0 → display feature button dialog; value1 - 23

F12: 0 → do quick search; value 1 - 23

13.9 Search buttons

The screen search buttons (displayed in figure 11-12) allows you to configure the search buttons.



Figure 13-12: configuration window 'search button'

13.9.1 Search button captions and hint

The names/captions of the search buttons in the Connecsy application can be changed in this window. These settings are user-dependent. They must be changed in the 'connecsy.ini' file in the language directory.

INI-Setting

Section: editCaptions
Key: Number of a DB field (value 0..11)
KeyValue: The name (caption) of the search button

For the hints it is the same procedure. The section in the INI-file is called: editHints

13.9.2 Location of database fields

When the search buttons in Connecsy have a caption you can map the database fields described in the DBData section to the defined edit fields. You can map the database fields with the combo box contents. The list of data fields (Data1 to Data12) is defined in the DBData section.

INI-Setting (Button contents)

Section: ButtonContents
Key: number of the search button (value 0 - 11)
KeyValue: number of the DB field you want to map the edit field

Example:

0=0

1=1

2=2
3=...

13.9.3 Edit field search definition

In this version of Connecsy it is possible when you type a search value in a field, the Connecsy application will automatically search in another database field. For example: The start position of the cursor is the edit field extension and the next edit field is person name. In this window you can configure that when you type a person name in the extension field, Connecsy searches in the extension field, but also in the person name field.

The number of database fields included in the query when the focus is on an edit field. The mapping is in relation to DBKeys section (explained earlier in this document).

Example:

0=1,2
1=2,1
2=3
.....
11=12

0=1 search on key 1 when searching is started in field 0
1=1,2 search on key 1 and 2 when searching is started in field 1

INI-setting(s):

Section: EditSearch

Key: number of the edit field

KeyValue: the number of the DB field, which would be included in a search query.

13.9.4 Calendar

To use the Calendar popup functionality, Connecsy must know which button corresponds with a Calendar popup. Check this box by the corresponding button/field that holds the Exchange name of the person so this person's Calendar will show when pushing that button. You can choose between Microsoft Outlook and Lotus Notes.

13.9.5 Email

When sending an email to the person, Connecsy must know which button corresponds with his/her email address. Check this box by the corresponding button/field that holds the email address of the person. When pushing the button, an email will be created which the email address already filled in.

13.10 Sheet contents

The configure window 'Contents' (displayed in figure 12-13) provides the settings for the search result sheet.

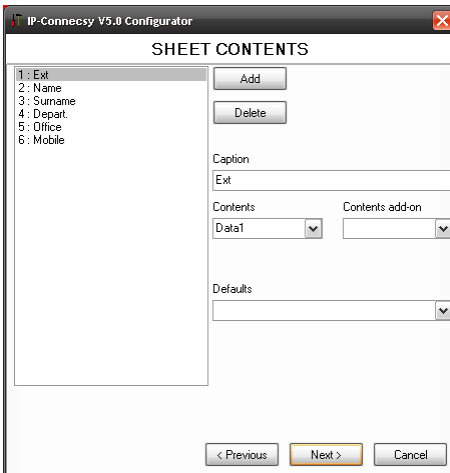


Figure 13-13: configuration window 'Sheet Contents'

With this sheet you can set how many fields you want to display in the result sheet in the Connecsy application. You can do this with the 'Add' and 'Delete' buttons. The configurator automatically uses the names of the search buttons. Filling another name at the parameter 'Caption' can change these names.

With the parameter 'Contents' you can map a database field to a sheet field (column). In one sheet field you can show two database fields. This can be set with the 'Contents add-on' combo box.

INI-Settings:

Section:	SheetContents
Key:	The number of the sheet column (value 0..12)
KeyValue:	The number of the DB field.

When there are two DB fields that are mapped to one sheet field (column) the data will be separated by a comma and a space character.

With the combo box 'Defaults' you can place the default name in a selected sheet field. The default names of the sheet columns are the names of the search buttons. These are set in the user-dependent ini-file with the section 'editCaptions'.

14. Hand made configurations

Not all settings are made via the Connecsy configurator. There are several settings, which must be set (if you want to change them), by editing the INI-files with a text editor. In this chapter the manual made settings will be explained.

14.1 Scroll Button Bitmaps

In the menu of the Connecsy application there is an option to show scroll buttons. These buttons have a bitmap on them. These bitmaps are programmable. They must be set in the connecsy.ini in the main Connecsy directory.

INI-Setting

Section: ScrollBitmaps

Key: Up/Down

KeyValue: the value should only contain bitmap names (no directories allowed), by default the bitmaps will be loaded from the subdirectory (bmps) in the Connecsy main directory)

14.2 F12 Search button

In the Connecsy configurator you can program the function button F12. In the dropdown list box you can choose the option quick search. For this key you can program a search with a predefined string on a predefined field.

INI-Setting

Section: F12SearchButton

Key: field

KeyValue: fieldnumber where to search (value 0 - 11)

Key: string

KeyValue: search string

14.3 Arrow key edit field order

With the 'arrow key edit field order' you can configure the order or move over the edit fields using the up and down keys.

INI-Setting

Section: Edit order

Key: numeric order 0 to 12

KeyValue: the edit field number

14.4 Character mapping table

In the connecsy.ini in the main Connecsy directory it has a character-mapping table concerning search strings. This section describes how the search string is composed internally. The defined table creates a mapping table for special characters.

For example: 130 = 101 means that the é (130) is treated like an e (101) in the search string. The result is that a name like 'René' is also found when your search string looks like 'rene' and vice versa.

INI-Setting:

Section:

Key:

KeyValue:

15. DEFINITY® settings

The next pages explain important DEFINITY® settings, which you must check in order to run the Connecsy application properly. Before you do this add the attendant(s) in the right way according to the normal DEFINITY® attendant settings. The DEFINITY® settings can be modified using a terminal emulation program TERRANOVA.

15.5 Attendant Type and Auto answer

Required form

Form	Field	Value
Change attendant <nr> (page 1)	Type	302B1
	Auto answer	none

Description

In this form you will find "Type" and the "Auto Answer" field. Type 'change attendant <nr>' at the TERRANOVA command prompt and check both fields (<nr> is the attendant number to check). The console type (302B1) is very important since this will identify the attendant console correctly as a Connecsy station to the DEFINITY®. If this is not correct the Connecsy application may produce unpredictable results. Only the Connecsy application where the "Console Type:" is principal, is able to activate the night-service. The "Auto Answer:" field must be "none". Figures 13-1 and 13-2 show example screen shots.

15.2 DXS/BLF

Required form

Form	Field	Value
Change attendant <nr> (page 1)	1: to 20: (Hundreds Select Button assignments)	Range value (see above e.g. 76)
Change console-parameters (page 2)	SAC Notification (Send All Calls Notification)	Y

Description

The hundred group select buttons can be used to enable the DXS/BLF consoles. This must be configured in the Definity®. Figure 13-1 shows an example screen shot. In case for instance the 7600 group needs to be monitored on the first DXS/BLF buttons the first entry should contain 76. In the Connecsy Configurator this must this 7600 (offset).

15.3 Feature buttons

Required form

Form	Field	Value
Change attendant <nr> (page 2)	1: to 24: (Feature Button assignments)	Featue (e.g. pos-busy)

Description

The feature buttons as defined in the "connecsy.ini" file should match the settings in the DEFINITY®. (see figure 13-3) The names of the feature buttons in Connecsy are only used as a reminder for the user, in case another feature button is programmed in the DEFINITY®, unpredictable results will occur. Some of the feature buttons are default connected with the function buttons of the Connecsy application, if those buttons are not programmed in the DEFINITY®, the specified function buttons in the Connecsy application will not work.

F2 is connected with feature button 1. This is a fixed connection.
 F5 is connected with feature button 6. This is a fixed connection.
 F7 is connected with feature button 10: last-numb
 F8 is connected with feature button 5: priority
 F11 is connected with feature button 24: pos-busy
 F12 is connected with feature button 23: night-serv

```
display attendant 1 Page 1 of 4
ATTENDANT CONSOLE
Type: 302 Name: Bedienpost
Extension: 1990 Group: 1 Auto Answer: none
Console Type: principal TN: 1 Data Module? n
Port: 500992 COR: 66 Disp Client Redir? n
Security Code: * COS: 14 Display Language: english
H.320 Conversion? y

DIRECT TRUNK GROUP SELECT BUTTON ASSIGNMENTS (Trunk Access Codes)
Local Remote Local Remote Local Remote
1: 2: 3: 4: 5: 6: 7: 8: 9: 10: 11: 12:
2: 3: 4: 5: 6: 7: 8: 9: 10: 11: 12:
3: 4: 5: 6: 7: 8: 9: 10: 11: 12:
4: 5: 6: 7: 8: 9: 10: 11: 12:

HUNDREDS SELECT BUTTON ASSIGNMENTS
1: 11: 5: 9: 13: 17:
2: 12: 6: 10: 14: 18:
3: 7: 11: 15: 19:
4: 8: 12: 16: 20:
```

Figure 15-1: Display/change attendant screens (page 1)

```
display attendant 1 Page 2 of 4
ATTENDANT CONSOLE

UIS FEATURE OPTIONS
Auto Start? y
Echo Digits Dialed? y

IP FEATURE OPTIONS
Remote Softphone Emergency Calls: as-on-local Direct IP-IP Audio Connections? y
Emergency Location Ext: 1990 IP Audio Hairpinning? y
```



Figure 15-2: Display/change attendant screens (page 2)

```
display attendant 2 Page 3
ATTENDANT CONSOLE
FEATURE BUTTON ASSIGNMENTS
1: split                13: class-rstr
2: dont-split           14: intrusion
3: goto-cover           15:
4: override             16:
5: priority             17: serial-cal
6: hold                 18: em-acc-att
7: trk-id               19: forced-rel
8: cdr1-alm             20: cw-ringoff
9: atd-qcalls           21: in-ringoff
10: last-numb           22: re-ringoff
11:                     23: night-serv
12:                     24: pos-busy
```

Figure 15-3: Display/change attendant screens (page 3)

15.4 Auto Start

Required form

Form	Field	Value
Change system-parameters features (page 5)	Auto Start	n

Description

The field "Auto Start" must be value "n". If the value is "y" the Connecsy application cannot handle both inbound and outbound calls correctly.

15.5 Calling Line Identification

Required form

Form	Field	Value
Display trunk-group <nr> (page 2)	Send calling number	Y
	Send connected number	Y

Description

To display the calling line identification, incoming calls must be routed through a listed directory number. Do not use the extension number of the attendant. This is the normal way of inbound call routing. If this is not correct then "Call from PC" will be displayed in the Connecsy console screen area for an inbound call. Check also the "Send Calling Number" field and the "Send Connected Number" field - both must be "y".

15.6 Time Remainder on Hold

Required form

Form	Field	Value
Change console-parameters (page 2)	Time Remainder on Hold	30 sec (normal)

Description

When a call is placed on hold, a timed reminder automatically alerts the attendant after an administered interval time. The attendant must re-enter the call before they can handle any other calls. To handle more than one call, the interval must be not too short. Change the interval timer and change the parameter of the field "Time Remainder on Hold (sec):" - normal 30 sec.

15.7 Hold-automatic

Required form

Form	Field	Value
Change System Parameters features (page 5)	Auto Hold	Y

Description

To put an active call on hold (without pressing the HOLD button, press a second call-appearance button. The second call appearance then becomes active.

Automatic Hold is the equivalent of pressing the HOLD button. A call placed in Automatic Hold is in Hard Hold.

Automatic Hold is disabled (the default), the active call appearance drops when a second call-appearance button is pressed. Default (n) in the Auto Hold field enables Automatic Drop.

15.8 Extension port type

Required form

Form	Field	Value
Change attendant <nr> (page 1)	Port, Security code	IP (for H.323), valid port for serial DCP.

Description

When using a serial DCP connection, the extension assigned to the attendant console should be assigned to a valid digital port. When using Connecsy over the H.323 IP-Console Protocol, the port type should be assigned to ip. When the port is configured as an IP port,

an optional password can be assigned to the port within the field Security code. See also figure 13-1.

15.9 Licenses

Every attendant logging on to the Definity® over the H.323 IP-Console Protocol requires a soft-console license. Every active connection uses one license from the pool of available licenses. When no more licenses are available, the Definity® will refuse any new logons. Make sure you have sufficient licenses available for all your operators. To make sure licenses aren't in use needlessly, log off from or close any inactive Connecsy console instance.

15.50 Enabling SNMP

Description

When using the BLF information to get an indication of presence of a person, it is very useful to have information whether persons are logged on or off their telephone sets. This feature is provided by Connecsy by the Connecsy Data Service which in turn requires an SNMP connection with Avaya Communication Manager. This section describes the configuration at the Avaya side which is needed to make this work.

Configuration Steps

Open a webbrowser like Internet Explorer or Firefox and enter in the addressbox the IP address of the Communication Manager. (This is the same IP address as it is used for ASA) Log in on the webinterface and then select the *Launch Maintenance Web Interface*.



Integrated Management
 Standard Management Solutions

[Help](#)
[Log Off](#)

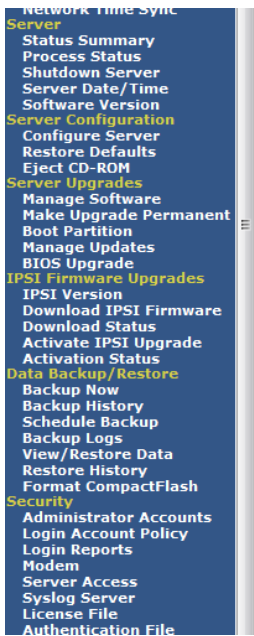
 Installation	Launch Avaya Installation Wizard The Avaya Network Region Wizard allows you to quickly administer network regions.	Launch Avaya Installation Wizard Launch Avaya Network Region Wizard
CM Administration	The Native Configuration Manager allows you to administer this system using a graphically enhanced SAT applet.	Launch Native Configuration Manager
Maintenance	The Maintenance Web Interface allows you to maintain, troubleshoot, and configure the media server.	Launch Maintenance Web Interface
Upgrade	The Upgrade Tool allows you to upgrade all servers, Survivable Processors, G700 Media Gateways, and G350 Media Gateways.	Launch Upgrade Tool

Then click in the left side menu within the section Alarms on SNMP Agents.



A configuration page appears and within this page select the option Access from *Any IP address*.

Also mark the check box for *Enable SNMP Version 2c*. Enter the community names as *public* and *private*.


[View G3-AVAYA-MIB Data](#)

Master Agent status: Up

IP Addresses for SNMP Access

☐ No Access

☒ Any IP address

☐ Following IP addresses:
IP address1 : IP address2 : IP address3 : IP address4 : IP address5 :

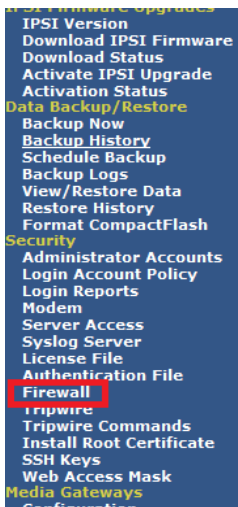
SNMP Users / Communities

☐ Enable SNMP Version 1
Community Name (read-only) : Community Name (read-write) :
☒ Enable SNMP Version 2c

Community Name (read-only) : public

Community Name (read-write) : private

The last step is to check the firewall of the server and if necessary configure it.
Click in the left menu the item Firewall.



Then mark the two check boxes next to *snmp*. Finally click Submit. At this point the Connecsy Dataservice can be started.

Input to Server	Output from Server	Service	Port/Protocol
☒	☒	ftp	21/tcp
☒	☒	ssh	22/tcp
☒	☒	telnet	23/tcp
☐	☒	domain	53/udp
☐	☐	bootps	67/udp
☐	☐	bootpc	68/udp
☒	☐	ttf	69/udp
☒	☒	http	80/tcp
☐	☒	ntp	123/udp
☒	☒	snmp	161/udp
☒	☒	snmptrap	162/udp
☒	☒	https	443/tcp
☐	☒	syslog	514/udp
☐	☐	ldap	389/tcp
☐	☐	ldaps	636/tcp
☐	☐	radius	1812/udp
☐	☐	secrID	5500/udp
☐	☐	safeword	5030/tcp
☒	☐	http-iphone	81/tcp
☐	☐	https-iphone	411/tcp
☒	☐	hp-ssh	2222/tcp
☒	☒	secure-sat	5022/tcp
☒	☒	def-sat	5023/tcp
☒	☐	echo-request	8/cmp

Submit

Advanced Setting...

Help

16. Connecsy PLUS

With the Connecsy PLUS you add extra functionality to the Connecsy V5.5. The Connecsy PLUS module provides the possibility to record a call.

Voice recording can only be used when Connecsy has been configured to the Serial DCP Media Connector or when the voice path is being established with a digital DCP phone when using Connecsy with the H.323 IP-Console Protocol. Voice recording is not supported on analogue, IP or external devices.

16.1 Automatic recording

Via the Connecsy menu you can configure the connecsy application to record calls automatically. When a line is active the application automatically starts recording for that specific line.

The recording stops when an active line becomes inactive. A line becomes inactive when a call is put on hold, when a call is transferred or when a line is released.

If a call is put on hold, the recording will stop. When the call is being resumed the recording will start again, but now the call will be put into another file. In this manner one call will result in several recorded files.

For the automatic recording you can configure the total number of saved recording files. The calls are recorded in a loop. When the total number of files is reached, the oldest recording file will be deleted. On this manner the disk capacity can be controlled. In the following table you see an indication of the disk capacity needed for recording.

The calls will be recorded with a transmission speed of 22050 bytes/sec.

Average Talk Time	Number of Calls	Capacity (MB)
2 min	10	26,5
2 min	15	40
2 min	25	66,2
5 min	10	66,2
5 min	15	99,2
5 min	25	165,4
7 min	10	92,6
7 min	15	139
7 min	25	231,5

It is also possible to save a record call so that this file will not be deleted in the automatic recording loop. You can secure a recorded call with the secure item in the recording form (see user manual).

You can recognise the secured recording files by a key item for a recording file.

Via the toolbar when you are in the recording form you can configure the total number of recorded files for automatic recording. You must select the item recording settings.

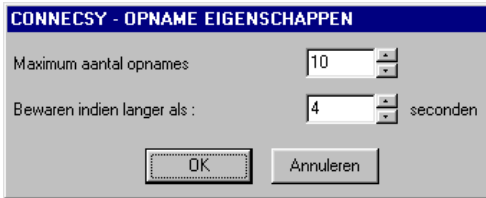


Figure 16-1: Record settings form

You can also configure the minimum talk time. When the calls are shorter than this talk time, the calls are not recorded.

The item Automatic in the menu Recording is secured with a password. Only the authorised persons can activate or deactivate the automatic recording. The password can be changed in the password form. By installing Connecsy the password is "connecsy".

16.2 Manual recording

An alternative for automatic recording is hand recording. When a line is active the recording can be started with the menu **Recording | start**. A manual recorded call is always secured and will not be deleted automatically. The recording stops when the line leaves it's active state (inactive). You can also stop the recording by the menu **Recording | stop**.

With the Connecsy configurator you can program a function button as a start/stop button for recording. See figure 15-2.

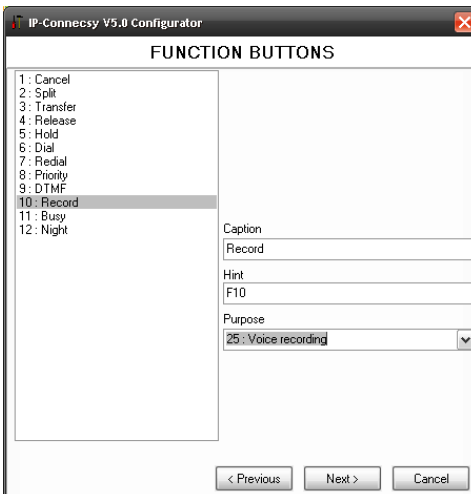


Figure 16-2: Configuration form function button for recording

When a line becomes the function button F10 will become active. You can now start recording a call by pressing the function button F10. After starting the recording, the record button will be changed to a stop button with which the recording can be stopped. Figure 14-3 displays the different states of the function button.

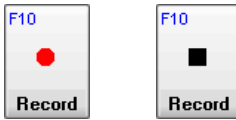


Figure 16-3: states of the function button record

16.3 Playing recorded files

Via the menu **Recording | play** the recording form will be displayed. The field names are also sort buttons. If a call of one person consists of several files the files can be sorted so the call is listed in order.

Choose the right file and press the button play. The Windows Media player will be started with the selected recorded call. Which playing application is started is configured in the connecsy.ini in the main connecsy directory.

16.4 Installation

The sound card in your Connecsy PC will record the calls. The voice will be extracted from the telephone line.

To do this you will need the Cybertech Digital Speech Converter (DSC100P.SA.DEF). The connection scheme is displayed in the following figure 15-4.

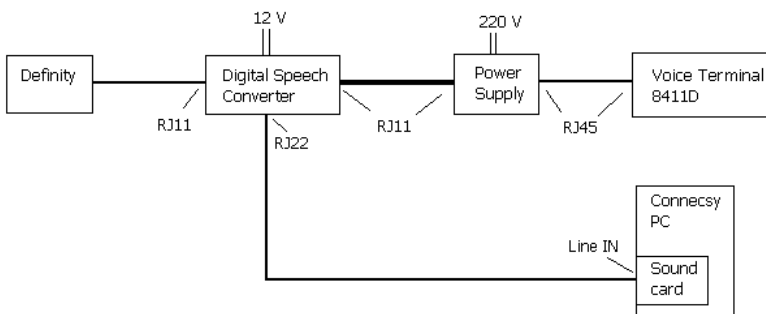


Figure 16-4: hardware connection schema for recording

The Digital Speech Converter must be included in the telephone connection of the 8411D. Use RJ11 connectors for this. These connectors are switched parallel. This cable is twisted (use the cable delivered with the converter).

In the connection of the 8411D a power supply is included. This must be included in the circuit after the converter.

At the other side of the converter one of the RJ22 connectors must be used the analogue signal for the sound card. These connectors are switched parallel. The two middlemost wires of the RJ22 connector are used for the analogue signal. The two outmost wires are used for the start-stop signalling.

The 2.5 mm jack connector passes the analogue signal to the sound card and is connected with the line-in input of the sound card. All conversations made with the 8411D voice terminal are passed to the sound card. With the Connecsy application you decide to record a call choosing either yes or no.

As the control of starting and stopping the recording of the analogue signal is performed by Connecsy, the Speech Converter Unit must be set to '**Constant Audio**' **modus**. The following steps should be taken:

Step	Action	Details
1	Connect a laptop or PC to the serial connection on the Speech Converter Unit using a 1:1 cable	9 pin D-Connector
2	Start Windows Hyperterminal to open a connection. A command line prompt will appear.	9600, N,8,1
3	Type the following command at the prompt:	key_audio all 1
4	Close the connection and see if Connecsy records any call correctly.	

The analogue signal is at the same time passed to the speakers. Mostly this is not desirable. This has to be switched off with the Windows mixer. Toggle the mute checkbox on of the line-in control (figure 15-5).

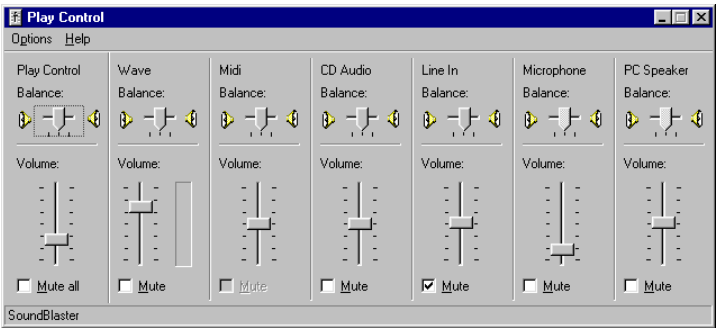


Figure 16-5: play control (mixer)

We earlier explained that you are able to configure the play application. You must configure this with a text editor. The settings are made in the connecsy.ini in the main connecsy directory.

INI-Settings

Section:	Settings
Key:	filecounter
KeyValue:	total number of (automatically) recorded files

Key: Automatic
KeyValue: 0 = hand; 1 = automatic

Key: Playcommand
KeyValue: for Win 98 it is:
`rundll32.exe amovie.ocx,RunDll /play / close`
For WinNT or later is this:
`mplay32.exe /play /close`

IMPORTANT: When the soundcard is not full duplex, you cannot hear ringing sounds, because the soundcard doesn't support recording and playing at the same time.

17. Connecsy LDAP

With the Connecsy LDAP you add extra functionality to Connecsy V5.5. The Connecsy LDAP module provides the possibility to import data from Directory Services like the one found within Windows 2000/XP/2003 Server.

LDAP (Lightweight Directory Access Protocol) is a protocol that enables an application to access a X.500 Directory Service by using a TCP/IP connection. With Connecsy 5.5 it is possible to do a database import using LDAP from such a Directory Service. The data can be imported both manually and automatically. The way the data from the Directory Service is being converted into the Connecsy database is defined in a Connecsy LDAP Profile-file.

17.1 Activating LDAP functionality

Entering an appropriate version code into the Configurator should activate the LDAP import functionality.

17.2 Manual import

If LDAP is enabled, there will appear a command in the File-menu labelled 'Import from LDAP'. By activating this command an 'Open File' dialog will appear. Select the Profile-file that is to be used and click Open. Then the update options can be selected (identical to an import from a CSV file) and the import will be started.

17.3 Automatic import

If LDAP is enabled, the dialog 'Automatic import' provides the possibility to pre-select a LDAP Profile-file for the import.

17.4 LDAP Profile-files

A Profile-file describes how the data from a LDAP Directory Service should be imported into the Connecsy database. A Profile-file is formatted as an .INI file.

[General]

The section **[General]** is required and contains the following keys:

signature:	should contain the value <code>\$LDAP_PROFILE\$</code> . This key is required.
version:	should contain the value <code>0.90</code> . This key is required.
description:	contains an optional description for the Profile-file.

[LDAP Session]

This section describes the parameters used to make a connection to a LDAP server:

Host:	contains the hostname or the host address of the LDAP server. Default value is <code>127.0.0.1</code> (localhost)
--------------	--

Port:	the port number of the LDAP server. Default value is the default LDAP port (389)
Transport:	states the transport of the connection: <code>TCP</code> or <code>UDP</code> . Default value is <code>TCP</code> ,
UserDN:	When the LDAP server requires authentication, this key contains the DN (distinguished name) of the user logging on. Leave this key empty if the server does not require authentication. Default value is empty.
Password:	Contains the password of the user named in UserDN . The password has been encrypted for safety purposes. Default value is empty.

[Mapping]

This section contains the parameters necessary to request the data from the LDAP server and how this data will be imported into the Connecsy database:

RootDN:	Contains the DN of the starting point (root) of each search action in the Directory Service. Default value is empty (this will be the 'root' from the Directory Service itself)
Scope:	Defines the method of searching the Directory Service started at the root: 0. Search on the level of the rootDN only; 1. Search on the first level below the rootDN; 2. Search on all levels below the rootDN, including the rootDN itself.
Query:	Defines the search condition, with which the Directory Service should be searched
Data1 t/m Data12:	Indicates the appropriate field of the Connecsy database. The value this field will obtain depends on the entered attributes or directive.

Queries

Queries should be entered in the RFC1558 specified format. The most simple query has a format of `(attribute = value)`, which will retrieve all entries in the database where the attribute will have the specified value, e.g. `(cn=*)` (This example will result in the LDAP server to return all entries, for which the Common Name attribute has a value) or `(cn=f*)` (will return all entries, for which the value of the Common Name attribute starts with a f).

Most of the LDAP servers have a limit for the maximum number of entries to be returned. When the search results exceed this limit, the server will send an error message. In this case no entries will be returned. In these situations, to be able to get valid results from the server, one can use a partitioned query. An example of a partitioned query is `(cn=[a-z]*)`. In this case Connecsy will create separate queries of `(cn=a*)` until `(cn=z*)`. The results of each individual query will be combined into one query.

Database fields

The keys **Data1** up to **Data12** indicate that the specific database field should be filled with the defined value:

```
Data[1-12] = attribute
attribute := attribute-name | directive
directive := $directive-type(parameter-list)
directive-type := SPLIT | CONCAT | LIST | VALUE
parameter-list := parameter [, parameter-list]
parameter := attribute | value | <empty>
```

Directive syntaxes:

```
$SPLIT(attribute, entries, separator)
$CONCAT(separator, attribute, attribute [,attribute]+)
$LIST(attribute, index)
$VALUE(value)
```

\$SPLIT

The **\$SPLIT** directive can split up the value of an attribute, using a specified separator and returning the value of the part indicated with entries. Entries have the following structure:

```
first-entry[-last-entry]
first-entry := F | f | numeric-value | <empty> (assumes F)
last-entry := L | l | numeric-value | <empty> (assumes L)
```

with which the numeric-value is zero-based.

F indicated the first entry (0), **f** the second entry (1), **L** the last entry and **l** the one before last entry.

Example: **\$SPLIT**(composer, -l, " ") where the data composer="Johan Sebastiaan Bach" will be returned as "Johan Sebastiaan" and **\$SPLIT**(composer, L, " ") will return "Bach".

```
sentence="The Quick Brown Fox"
$SPLIT(sentence, F, " ") -> "The"
$SPLIT(sentence, 1-l, " ") -> "Quick Brown"
$SPLIT(sentence, L, " ") -> Fox
```

\$CONCAT

The **\$CONCAT** directive combines the specified list of attributes to one value and divides these attributes with a separator:

```
part1="is"
part2 = "This"
part3 = "separated"
$CONCAT(":", part2, part1, part3) -> "This:is:separated"
```

\$LIST

The result of an attribute can consist of more than one value. Default the first value is always returned. The **\$LIST** directive can be used to select a specific item from the list. A item is selected by using a position index (0-based) to point at the specific value in the list of values.

Example:

```
list="Value 1", "Value 2", "Value 3", "Value 4", "Value 5"
$LIST(list, 3) -> "Value 4"
```

\$VALUE

The **\$VALUE** directive returns the value parameters as absolute value.

Example:

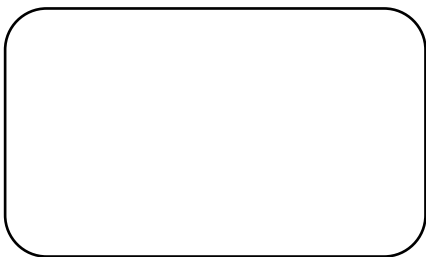
```
$VALUE("This is a value") -> "This is a value"
```

Nested directives

When the parameters of a directive require an attribute, another directive can be used:

```
$SPLIT($CONCAT(" ", $VALUE("This"), $VALUE("is"), $VALUE("a nested"),  
$VALUE("directive")), 1-l, "s") -> " a nested"
```

Your dealer:



AVAYA
AVAYA